The Dissertation Committee for Edward Joseph Artnak III Certifies that this is the approved version of the following dissertation:

Development and Assessment of CFD Models Including a Supplemental Program Code for Analyzing Buoyancy-Driven Flows Through BWR Fuel Assemblies in SFP Complete LOCA Scenarios

Committee:

Steven R. Biegalski, Supervisor

John R. Howell

Karl W. Schulz

Mark R. Deinert

Alexandre K. da Silva

Development and Assessment of CFD Models Including a Supplemental Program Code for Analyzing Buoyancy-Driven Flows Through BWR Fuel Assemblies in SFP Complete LOCA Scenarios

by

Edward Joseph Artnak III, B.S.M.E.; M.S.E.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

December 2012

"...neither theory nor experiments alone will ever solve complex problems of fluid flows — a combination of both is needed."

— Learned Committee, Academie des Paris, 1775.

Dedication

To my parents, Joe and Katie Artnak, and loving wife, Nicole, for their unwavering encouragement and support throughout this ambitious research undertaking in pursuance of a Doctor of Philosophy in Mechanical Engineering.

Acknowledgements

I would first like to thank Dr. Randall Gauntt from Sandia National Laboratories for briefly assimilating me into the Reactor Modeling and Analysis Group he supervises and for having the confidence in my ability to individually explore such a challenging research topic. In addition, the experimental insight and guidance of Dr. Eric Lindgren and Dr. Kenneth Wagner were instrumental in the development of computational models and simulations that best replicate the physical experiments performed at Sandia National Laboratories. This work would not have been possible without the collaborative effort and support of these individuals, Sandia National Laboratories, and the United States Nuclear Regulatory Commission.

I would also like to thank my supervisor, Dr. Steven Biegalski, and members of my dissertation committee that include Dr. John Howell, Dr. Karl Schulz, Dr. Mark Deinert, and Dr. Alexandre da Silva for their patience, direction, and giving of their time to facilitate a positive result in my pursuit of a Doctor of Philosophy in Mechanical Engineering. A special thanks to Dr. Howell is also warrant for initially recommending and encouraging the utilization of HPC resources available through the Texas Advanced Computing Center (TACC) in the furtherance of computational modeling objectives undertaken as part of this research endeavor.

Furthermore, the computing as well as visualization resource requirements for project completion were immense and accordingly I would like to thank Dr. Karl Shulz for his time and effort in making these resources available through the TACC while asking nothing in return. The computational achievements resultant from this research project were made possible only through the countless contributions of Dr. Schulz

including the installation, setup, and debugging of numerous versions of commercial CFD software as well as personal job execution and control instruction for multiple high performance computing systems.

Most importantly, I would like to thank Dr. Steven Biegalski for granting me a truly rare opportunity to continue my education here at The University of Texas at Austin and for initiating financial support through a graduate research appointment to do so. Of special note is the amazing ability of Dr. Biegalski to balance his professional role as mentor, guide, and advisor with a genuine personal effort to establish and maintain meaningful student relationships - for his friendship I am especially grateful. The devotion of time and loyalty to students within the department is second to none.

Lastly, but only because it is revealed without words every day, my most heart-felt appreciation to Nicole for her unyielding support, encouraging words, and enduring patience for so willingly sharing a truly arduous graduate lifestyle.

Development and Assessment of CFD Models Including a Supplemental Program Code for Analyzing Buoyancy-Driven Flows Through BWR Fuel Assemblies in SFP Complete LOCA Scenarios

Edward Joseph Artnak III, Ph.D.

The University of Texas at Austin, 2012

Supervisor: Steven R. Biegalski

This work seeks to illustrate the potential benefits afforded by implementing aspects of fluid dynamics, especially the latest computational fluid dynamics (CFD) modeling approach, through numerical experimentation and the traditional discipline of physical experimentation to improve the calibration of the severe reactor accident analysis code, MELCOR, in one of several spent fuel pool (SFP) complete loss-of-coolant accident (LOCA) scenarios. While the scope of experimental work performed by Sandia National Laboratories (SNL) extends well beyond that which is reasonably addressed by our allotted resources and computational time in accordance with initial project allocations to complete the report, these simulated case trials produced a significant array of supplementary high-fidelity solutions and hydraulic flow-field data in support of SNL research objectives.

Results contained herein show FLUENT CFD model representations of a 9x9 BWR fuel assembly in conditions corresponding to a complete loss-of-coolant accident scenario. In addition to the CFD model developments, a MATLAB based control-volume model was constructed to independently assess the 9x9 BWR fuel assembly under similar accident scenarios. The data produced from this work show that FLUENT

CFD models are capable of resolving complex flow fields within a BWR fuel assembly in the realm of buoyancy-induced mass flow rates and that characteristic hydraulic parameters from such CFD simulations (or physical experiments) are reasonably employed in corresponding constitutive correlations for developing simplified numerical models of comparable solution accuracy.

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

1.1 Introduction to Problem

MELCOR, a light water nuclear reactor accident analysis code, was developed by Sandia National Laboratories (SNL) as a fully integrated, quick response numerical modeling tool that is capable of predicting severe reactor accident outcomes based on the likelihood of progression with which these accidents occur. These characteristic accident progressions modeled by the MELCOR software include but are not limited to the following considerations: (1) thermal-hydraulic response in coolant systems, core, and containment; (2) core deterioration and heat-up; (3) radionuclide release and transport; and (4) related physical phenomena expected in a severe accident scenario. Users have an ability to exert some control over the program's response through inputs called sensitivity coefficients to tailor the system assessment; however, other hardwired geometry models limited within the program impose certain user restrictions (Gauntt *et al.*, 2000). When considering an objective to improve code performance and accuracy, the latter (hardwired models) require a significant increase in effort and solver modifications to implement corrections that, as will be discussed briefly, is beyond Sandia National Laboratories' intended scope of consideration at this time.

The importance of contributing vital simulated outcomes to overwhelming and analytically difficult problems postulated from severe loss-of-coolant circumstances surrounding an accident, ordains this unique software as a primary assessment tool of the Nuclear Regulatory Commission (NRC) in modeling such scenarios (Reinert, 2004). It is also used to instill decommissioning safety regulations and guidelines, including an expansion of related accident analyses to spent fuel pools (SFP) of operating nuclear

power plants. Continuing efforts to improve upon the computer code's accuracy, accident progression capability, and foresight into unlimited accident initiators reinforces the NRC's commitment to establish a meaningful and mutually beneficial relationship between commercial nuclear power and public safety.

As part of this on-going commitment and in response to prior recognition of overly simplified assumptions of a conservative nature from earlier evaluations of potential accident risk in a spent fuel pool [NUREG-1738] (Gauntt and Wagner, 2004), an extensive modeling effort was undertaken to assess the intrinsic consequences of utilizing the preceding simplified methods of analysis. Under severe conditions similar to those introduced in a 2001 risk assessment study performed by NRC staff involving a typical decommissioning plant, the resultant SFP accident response was simulated to describe potential feedback from a reference boiling water reactor (BWR) SFP subject to a complete loss-of-coolant scenario utilizing the MELCOR 1.8.5 software. Figure 1.1 shows the referenced pre-accident configuration of the entire reactor building and spent fuel pool. The concluding report presents findings from a thorough (simulated) investigation into a number of limiting variable analyses and the ensuing overall (interdependent) system response due in part to the variances in these parameters (Gauntt and Wagner, 2004).

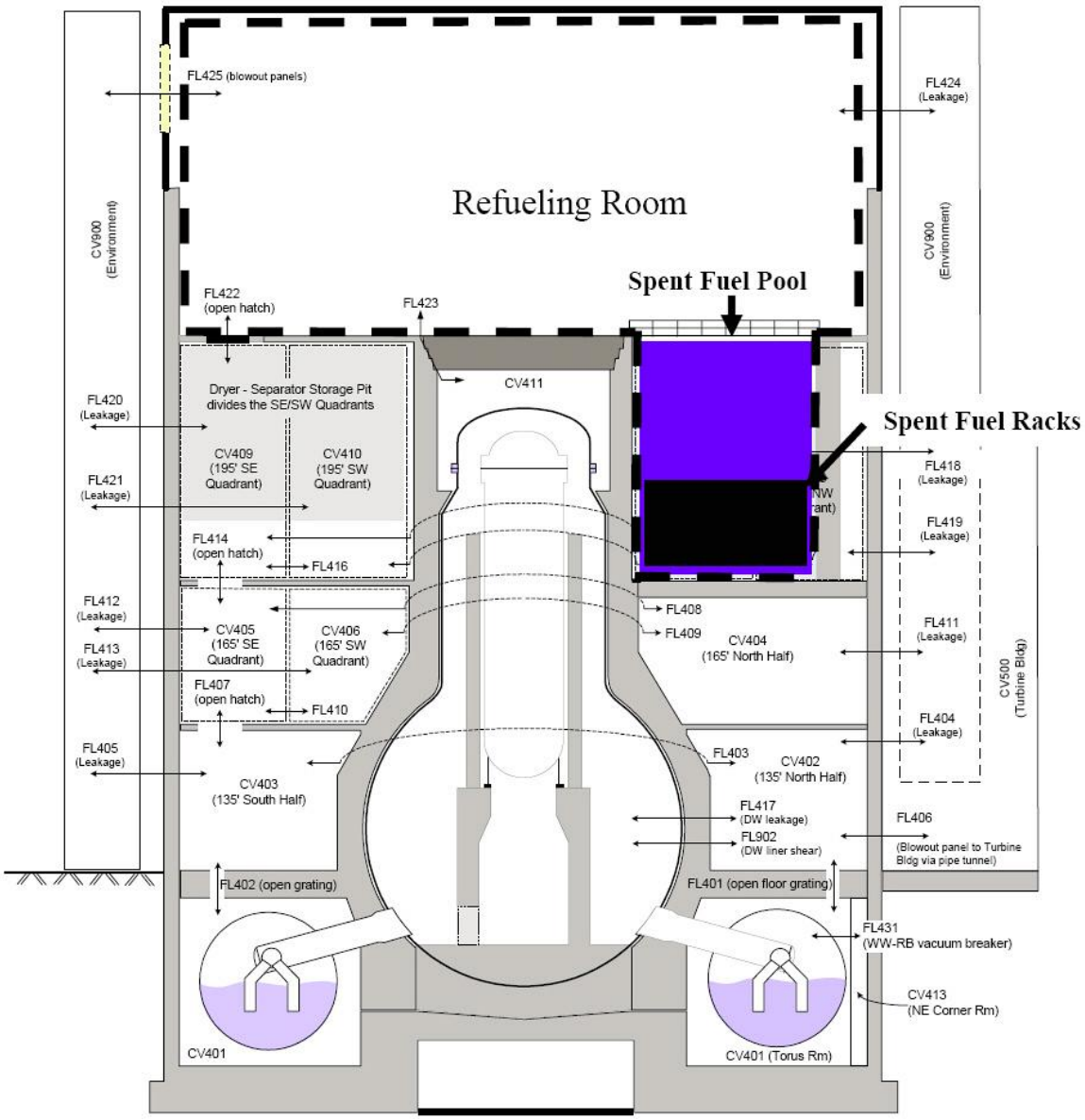


Figure 1.1: Reference reactor building for a boiling water reactor and spent fuel pool (Gauntt and Wagner, 2004).

As a result of inherent assumptions embedded within MELCOR's initial fuel response models, the simulated solutions revealed in this report ultimately depended upon several preceding aside calculations in fluid dynamics and parametric analysis to develop a suitable MELCOR SFP model capable of simulating a SFP accident domain. The

adaptations transforming the original fuel model were a critical requirement prior to the model's implementation of numeric trials, which produced the final results. Concerns subsequently arose regarding the physical basis and interpretation of implied assumptions on which the new MELCOR SFP model computer code was founded to obtain the simulated outcomes. A majority of these concerns were in fact clearly identified and described in the same concluding report, including the need to consider large variations in geometric length scales, various contributions (both independently and dependently) affecting heat removal rates, flow patterns, and thermal coupling among many other considerations for an accurate analysis. These concerns ultimately provided motivation for the NRC to fund a second experimentally rooted effort to explore the variable effects in related behavior-defining parameters to the aforementioned phenomena in a more accountable, but computationally manageable, manner.

Ideally, the end goal is an accurate representation of these parameters as a culmination of one or more combined variable values (if not a complete replacement) with accurately measured quantitative data. The importance of deriving accurate correlations among groups of parameter values here stems from a need to imply modeling assumptions (or limit them to the most likely or widely applicable conditions) within the MELCOR software considering the vast array of physics modeled on a larger scale by the code itself. The individual evaluation of parameters at every scale level is impractical and debilitating to overall solver performance.

Intrinsically, like many other control volume codes, MELCOR relies heavily on constitutive relationships to dictate system response as in the viscous flow losses (also known as wall shear stress or friction losses) and the inertial flow losses (also known as form or minor losses) across control volumes via what MELCOR terms flow paths (Durbin and Lindgren, 2005). A similar conclusion may be drawn in regard to the

treatment of thermal radiation heat transfer among grouped SFP assembly elements and the implementation logic governing the response of MELCOR's radiation heat transfer model. Two examples are purposely illustrated here for clarity; however, only the former warrants attention concerning the current research.

In summary, a need is presented to provide basic thermal hydraulic data in accordance with a SFP complete loss-of-coolant accident to enhance and/or facilitate code validation of the MELCOR SFP model and reduce concerns associated with the interpretation of any experiments. Ideally, this serves to establish a system of inherent checks and balances for solidifying accurate results among original analytical, experimental, and computational study. The thermal hydraulic data provides critical information needed to accurately calibrate parameter values and account for viscous and inertial resistances within individual flow path segments. These losses work against the buoyancy-driven convection induced by temperature gradients and are extremely important precursors to the type of transient thermal behavior and nature of heat propagation (notably the spread of an initiated zircaloy cladding fire). Undoubtedly, this information will influence future mitigation guidelines for fuel assembly management (Lindgren, 2004) in typical spent fuel pool configurations.

In an effort to fulfill the above-stated need, Sandia National Laboratories uses a joint computational and experimental effort to provide essential data and analysis for the improvement and validation of the MELCOR SFP model. This is the fundamental motivation behind larger efforts from which current research focus and objectives are derived.

1.2 LITERATURE REVIEW

The volume of available published work specific to the utilization of FLUENT Incorporated's commercial FLUENT® computational fluid dynamics (CFD) software (FLUENT Inc., 2005), in any facet to analyze thermal fluid flow behavior in boiling water reactor fuel assemblies, is quite simply limited as one might conclude from the rather narrow field of study. The regularity in which the related material is considered either sensitive or proprietary also prevents widespread dispersion and accessibility of research specific to this area. Attempts to obtain published analytical studies pertaining to buoyancy-driven airflow through a SFP fuel assembly, a circumstance that inherently constitutes a problematic scenario, culminates with the same conclusion. Furthermore, it is important to remember that modern CFD analysis as we define it today, including the capability to model in three dimensions, has only become a practical and reasonable expectation in the last fifteen years with the availability of high performance computer (HPC) workstations at a reasonable cost.

Despite the lack of abundant reference literature specific in scope to the project at hand, a range of supporting published research is available from which valuable insights and/or key developments derived in part from an analytical or experimental study are applicable to areas/issues relevant to the current research. Similarly, contributions from unpublished literature studies, guidelines, and manuals are widely available and heavily utilized as sources of supporting information and data. Due to the large volume of supporting reference literature, only a brief description of the context or premise from the most relevant sources or applied contributions to this research is afforded in the following review.

1.2.1 General CFD Development

It is well known that the fundamental foundation of CFD is derived from the basic governing equations of fluid dynamics. The historical evolution of fluid dynamics can be traced back to the seventeenth century where the development of experimental fluid dynamics began to take form. This was followed by the gradual progression of theoretical fluid dynamics in the eighteenth and nineteenth centuries, expanding through the introduction of accurate numerical algorithms and high-speed digital computers in the latter twentieth century to include the modern practice of CFD. Therefore, the whole discipline of fluid dynamics can be considered as the study and development of three constituents: pure experiment, pure theory, and CFD (Anderson, 1995). Historic accounts of the early contributors and advancements in the field of fluid dynamics leading up to modern CFD are detailed in the literature elsewhere (Tokaty, 1971; FLUENT Inc., n.d.). Published works addressing the fundamental principles governing the physics of fluid flow behavior and thermal fluid heat transfer with correlations between theory and analytical equations also exist and are widely available (Alexandrou, 2001; Bird *et al.*, 1960; Incropera and DeWitt, 2002; Fox and McDonald, 1985, 1992; Holman, 1990; Young *et al.*, 2004).

Beginning in the 1960s, substantial research efforts focused on the development of numerical methods and complex computer algorithms that are still implemented in CFD codes today. There are numerous published texts on the subject, but discussions on the fundamental aspects of the governing equations, numerical discretization techniques, and applications in CFD (Anderson, 1995) as well as specific consideration toward the development and implementation of the finite-volume method (FVM) CFD technique in commercial software (Versteeg and Malalasekera, 1995), are of particular relevance to this research project. These books offer an invaluable insight into the general sets of

complex nonlinear mathematical expressions and solution algorithms imbedded within typical CFD software programs.

A second body of contributing literature includes materials that specifically pertain to the functionality, methodology, or applicability of the FLUENT CFD modeling software. This consists of validation guides, training and tutorial exercises, standards and guidelines articles, user guides, modeling guides, advanced training modules, and other supplemental information accompanying or written explicitly in consideration of the FLUENT CFD software (Bell, 2004; Bhaskaran and Collins, 2002; Chilka and Kulkarni, n.d.; FLUENT Inc., 2005, 2006; Iaccarino, 2004). The listed references are just a sampling of the more informative pieces of supplemental literature reviewed and factor into the quality of research performed.

Other noteworthy developments in the analytical modeling of low-Reynolds-number (low-Re) viscous flows that are relatively recent include an improved version of Wilcox' original 1988 k-ω model containing a refinement for low-Re (transitional) flow effects (Wilcox, 1998) and an entirely new model proposed for boundary layer transition using a single-point Reynolds-averaged Navier-Stokes (RANS) approach (Walters and Leylek, 2004). The models described in these papers provide a basis for the recently integrated standard k-ω model and k-kl-ω (beta) model in FLUENT, respectively. Although the developing flows investigated in this research are well-within the laminar flow regime (as discussed in subsequent sections of this report), quantifying any potential pressure-loss contributions due to transitional flow effects (no matter how minute) is of significant importance.

1.2.2 Early Analytical Analyses and Models

During the ten-year span from 1975 to 1985, a number of independent research efforts were undertaken in an attempt to develop more accurate analytical methods and models for analyzing possible loss-of-coolant accidents (LOCA) at operating reactors. Increasingly severe design criteria associated with high-density re-racking of spent fuel to accommodate growing inventories as well as potential faults leading to drainage of coolant water or circulation failure in a reactor or storage pool were the primary motives for these early endeavors. Details concerning the consequences of these critical events and other accident initiators have been discussed in a recent report (Alvarez *et al.*, 2003).

Despite a common research focus, proposed methods to analytically evaluate or model a light water reactor (LWR) fuel bundle under LOCA conditions are largely dissimilar in their computational approach, scope of physics considered, fidelity of thermal hydraulics represented in governing equations, and intrinsic rigorousness of applied assumptions. Several of these research efforts predominantly focused on in-core fuel assemblies under LOCA or emergency core cooling system (ECCS) operation conditions where radiative heat transfer was assumed significant in comparison to conduction and convection alone. Under these conditions, heat removal occurs through two primary radiant heat exchanges: (1) between two or more solid surfaces (*i.e.* fuel rod cladding, shroud, and other subchannel walls) and (2) between solid surfaces and the coolant (*i.e.* water vapor and droplets in the two-phase state). Initial issues concerning conventional models and their methods of evaluating the former radiant heat transfer mechanism during a LOCA were the development focus of the CIDER computer code, which added the capability to handle circumferential temperature gradients on fuel rod surfaces as well as non-uniform radiant heat flux (Naitoh *et al.*, 1977). A study investigating the core spray cooling ECCS for a BWR addressed the latter radiant heat

exchange mechanism through the development of a unique model to evaluate analytically the heat transfer coefficient during spray cooling, accounting for both radiation and convection to the two-phase water vapor-droplet medium (Sun *et al.*, 1976).

Unfortunately, like most early codes which emphasize only one aspect in the entire radiant heat transfer phenomenon while simplifying or neglecting the others (Mandell, 1979, 1981), these two models do not afford equal consideration of both radiative heat exchange mechanisms which are assumed important to the fuel response modeling during LOCA conditions (as stated in both relating articles). Proposed methods for assessing other aspects of radiant heat transfer in rod bundles exist elsewhere (Gotovskii *et al.*, 1984). Subsequent research attempted to alleviate the single-aspect emphasis associated with these models by developing a more versatile, yet computationally efficient method of calculating radiant heat transfer during LOCA conditions in comparison to the CIDER computer code implementation. Consequently, the formulation of a matrix calculation method for radiative heat transfer was derived and applied within several simple model variations for analyzing heat exchange in a LWR fuel bundle (Uchida and Nakamura, 1981; Lorenzini *et al.*, 1983).

While the aforementioned analytical calculations and proposed modeling schemes offer a uniquely different array of possibilities to consider with regard to modeling techniques and assumptions applicable to specific LOCA conditions, the models are still in fact primitive despite clever implementations. The inability to function as standalone models in the analysis of hypothetical accident scenarios stems from the absence of a thermal hydraulic approach for concurrent evaluation of buoyancy-induced flow intricacies. The lack of an explicit solution (due to the interdependence of governing heat transfer relationships) restricts the applicability of these models to use as subcodes in

more general fuel assembly analysis programs unless numerous limiting assumptions and parameter values can be employed.

In addition, studies investigating the effect of thermal radiation on fuel assembly temperature distributions in severe accident scenarios subject to complete loss-of-coolant without convection support the supposition that direct modeling of radiant heat exchange is unnecessary in certain LOCA modeling analyses (Artnak, 2008; Reinert, 2004). As discussed in subsequent sections, the effect of thermal radiation over the course of a transient heating period is characterized in the reports as mollifying the temperature gradient between the fuel rods in such a way that convective heat transfer is maximized by the redistribution of temperatures across the fuel bundle. Therefore, the effect of thermal radiation can be indirectly modeled with reasonable accuracy by simply enforcing a similar boundary condition at the fuel rod surfaces as shown to develop from the effect of radiant heat exchange in convection-limited fuel assemblies (*i.e.* uniform fuel rod surface temperatures or heat flux at specific height ranges along the axis). However, the validity of this assumption is restricted to buoyancy-driven convective heat transfer under complete loss-of-coolant conditions in which the coolant fluid is substituted with a single-phase air medium.

A direct consequence of this understanding is the significance attributed to the buoyancy-driven heat removal aspect of the thermal hydraulic phenomenon as a primary influence in predicting the temperature response of a fuel bundle during a complete loss-of-coolant accident. Early code developments directed toward the evaluation of natural circulation within fuel assemblies predominantly existed as thermal hydraulic methods of analysis for the licensing of spent fuel storage pools. NUS Corporation developed the SFPT computer program (Renner, 1976) for this application, which simulates one-dimensional buoyancy-induced flow through a single loop comprised of heated flow

paths representing spent fuel assemblies, and one connected return flow path representing the downcomer between the storage rack and pool wall (Gay, 1984). A few similar one-dimensional models devised from the basic SFPT analytical method with slight differences in the applied assumptions are presented in the literature elsewhere (Niyogi and Tseng, 1983; Singh and Soler, 1983). The SFPT model utilizes an iterative solution procedure in order to calculate the exit temperature of every assembly flow path given the heat generation rate in each, which subsequently determines the density difference and resulting buoyant driving force. The pressure differential is equated to the buoyant driving force through lumped treatment of hydraulic flow losses across simplified flow paths, which permits iteration on the mass flow rate to obtain a steady-state solution.

Improvements over the initial one-dimensional analytical models are evident in the development of another computer code, GFLOW, which offers the added capability to predict three-dimensional (3D), transient, natural circulation flows in a SFP and includes a radial heat conduction routine for calculating fuel rod temperature distributions (Gay, 1984). The use of finite-difference methods to calculate independent variable values at designated nodes allows for increased geometric detail in the model analysis compared to the one-dimensional formulations mentioned above. However, GFLOW is specifically designed for modeling accident SFP conditions characterized only by a failure to circulate the water coolant inventory. Therefore, the code incorporates the assumption of constant material properties for the coolant fluid (*i.e.* density, coefficient of expansion, thermal conductivity, viscosity, and specific heat are the same everywhere in the SFP and do not vary with time). Although this is a reasonable assumption under intended application conditions, the accurate consideration of temperature dependent air properties in a complete loss-of-coolant severe accident is critical. In addition, both the SFPT and GFLOW models require user inputs for a number of hydraulic loss coefficients

that are typically unknown in postulated SFP accidents, yet significantly influence the temperature response in buoyancy-induced convective airflows.

A final culmination of noteworthy research efforts conceive experimental analyses and computer modeling codes with the greatest relevance to SFP complete loss-of-coolant accident scenarios consistent with the focus of this project. The SFUEL computer code, developed by SNL, provided the capability to analyze the thermal hydraulic phenomena associated with complete drainage of SFP water and included formulations for energy transfer by mechanisms of conduction, convection, and radiation (Benjamin *et al.*, 1979). A subsequent variation in the SFUEL series of codes, SFUEL1W, added improvements for handling self-sustaining zirconium-air oxidation and propagation in spent fuel assemblies (Best and Pisano, 1983), accompanied in development by physical experimental studies of rapidly oxidizing zircaloy cladding in fuel bundles (Stalker and Benjamin, 1983). The newer SHARP program code, written by Brookhaven National Laboratory (BNL), intended to provide the same assessment of thermal hydraulic characteristics as the SFUEL series of codes but utilized a simplified analysis method to model specific SFP layout configurations for fuel heat-up calculations (Travis *et al.*, 1997).

Extensive work has employed these computer codes in the investigation of spent fuel heat-up during complete loss-of-coolant conditions for gauging levels of potential risk. The details surrounding several significant experimental analyses performed using these models including associated results, ill-posed assumptions, and existing modeling code deficiencies inherent to the calculations are discussed in a recent report (Collins and Hubbard, 2001). However, issues regarding accuracy of the data upon which empirical relations within the programs are based and the inability or neglect to include flow losses from the grid spacers, flow mixers, and tie plates within the models are particularly

troublesome. The COBRA-SFS computer code, developed at Pacific Northwest National Laboratory (PNNL), eliminates some of the uncertainties associated with these codes by offering finer resolution of the energy distribution and flow field using a finite-volume subchannel modeling approach, which allows complex three-dimensional geometries to be more accurately represented (Rector *et al.*, 1986). This thermal hydraulic computer code was originally designed for precise analysis of complex spent fuel dry storage systems to predict temperature and velocity distributions under mixed and/or natural convection conditions, but significant efforts to assess the validity of the code have demonstrated applicability to a wide variety of systems (Lombardo *et al.*, 1986).

Despite these improvements, the COBRA-SFS program code still implements a lumped-volume/parameter, quasi three-dimensional approach that relies on some form of mathematically characterized parameters for geometric representation. Therefore, a number of limiting assumptions and uncertain correlations are unavoidable implementations within the code. In modeling fuel assemblies under complete loss-of-coolant conditions where buoyancy-driven convection is a critical heat transfer mechanism, user input requirements for flow resistance correlations and loss coefficients are extremely solution-sensitive parameters that are usually unknown. These attributes severely hinder the certainty of solutions obtained using the COBRA-SFS computer code (or any other code discussed) for scenarios of current interest to this project.

In summary, all of the aforementioned analytical methods, proposed models, computational codes, and associated experimental studies are critically deficient in one or more aspects involving the accurate assessment of temperature and airflow rate response within a fuel assembly under a complete loss-of-coolant setting, where the principal heat removal mechanism is buoyancy-driven natural convection. The capability of prominent commercial CFD software to incorporate true geometrical representation at a high

fidelity, three-dimensional integral analysis using the FVM CFD technique, and limited assumption computation within a model offers significant benefit in comparison to other simplified approaches. Again, this is especially substantial in terms of eliminating user input requirements for loss coefficients, which are extremely solution-sensitive and ascribe considerable uncertainty to analyses using explicitly defined values.

1.2.3 Recent CFD Research in Nuclear Applications

To date, the use of modern commercial CFD software for high-fidelity, exact-geometric modeling of airflow through a prototypical BWR fuel assembly remains absent from the collective research presented in open science literature publications (to the best of the author's knowledge). However, the volume of available work encompassing efforts to utilize commercial CFD software in a loosely similar manner, geometry, or field of nuclear application to the current research spreads across a considerably wider range and number of publications. In recent years, the most extensive CFD models developed for related nuclear application analysis are generally credited to one or more research efforts from a national laboratory, nuclear component manufacturer, or CFD software development corporation. These significant, well-presented computational models are characterized by extremely high-number element meshes for accurate geometric representation of the domain with high-fidelity analyses including: (1) pressurized water reactor (PWR) core cooling simulation utilizing a 240 million-cell mesh to model thermal hydraulics in one-eighth of the core geometry with porous media representation of spacer grids (CD-adapco, 2000); (2) detailed 13.5 million-cell model of flow through a 5x5 subchannel section of a PWR spacer grid (CD-adapco, 2000); (3) validation case study of a 5x5 rod bundle with concept grid representing a PWR

subchannel section of a single-span using a 20 million-cell mesh (Conner *et al.*, 2010); and (4) simulation of flow distribution in a partial length Advanced Burner Test Reactor (ABTR) benchmark geometry consisting of a helical wire-wrapped 19-pin fuel assembly with 85.5 million-cell mesh representation (Hamman and Berry, 2009).

The examples above represent a few of the most extensive, recent research efforts utilizing commercial CFD software for conducting computational experiments, but these and other equivalent high-fidelity models are few in number. Various other attempts have utilized lower-resolution models with coarse mesh representation of the domain geometry and/or inclusion of minimal subchannel section geometry with limiting assumptions in performing similar computational experiments to the aforementioned. These research works include small subchannel and spacer section models of related PWR geometries (Toth and Aszodi, 2010; Liu and Ferng, 2010), coarse mesh models of similar 5x5 rod bundle and spacer representations of a single-span PWR subchannel segment (Ikeda *et al.*, 2006), and simple mesh models of wire-wrapped rod bundle subsections relative to advanced reactor fuel designs (Gajapathy *et al.*, 2007; Natesan *et al.*, 2010). It is important to note that implied notions of lower resolution, coarse mesh, or simple model in the above analyses is not necessarily a direct correlation to decreased solution accuracy.

On a different scale, research studies have also employed commercial CFD software to develop models for simulating larger nuclear component and reactor subsystem domains than previously considered, although these analyses require additional modeling assumptions. Recent articles detail CFD modeling solutions from analyses of flow distributions in the downcomer and lower plenum of several PWR designs (Hohne *et al.*, 2006; Rohde *et al.*, 2006; Jeong and Han, 2008), thermal hydraulic response in a material test reactor (MTR)-type fuel assembly composed of three simple

geometric regions (Aglar *et al.*, 2007), and thermal fluid flow in a typical spent fuel storage cask system (Lee *et al.*, 2009). Several additional research efforts of particular interest to the current project focus on SFP accidents include computational experiments employing CFD models for investigating velocity and temperature distributions in a large SFP containment building during a LOCA scenario (U.S. Nuclear Regulatory Commission, 2001, Fall 2001). The current use of CFD software in general nuclear reactor analyses and identified obstacles requiring resolution before sufficiently dependable, accurate solutions are obtainable for safety design studies are discussed in a recent seminar (Weaver *et al.*, 2003).

Despite arguable dissimilarities in comparison to the current project scope of research, the above-mentioned works exhibit some degree of congruency as leading development efforts to incorporate commercial CFD software into the nuclear field of application. However, there remains a clear need to assess the potential for high-fidelity CFD models and computational simulations to provide accurate, meaningful results in analyzing the thermal hydraulic response of a BWR fuel assembly under complete loss-of-coolant conditions; especially the prototypic experimental assembly in question.

1.2.4 Simple Methods for Modeling Buoyant Flows

Although CFD models are capable of providing accurate solutions to analyses of very complex systems with a minimal number of assumptions, the method incurs a significant cost in the form of model development effort, the dedication of a large computing resource, and an extensive CPU-hour time increment for completion. As a result, the frequently sought benefit of conducting a limited number of CFD computations is obtaining values for critical parameters and/or data for empirical

correlations necessary in the development of (or required as input to use) a simplified, efficient model that affords relatively accurate solutions to the same system under perturbing conditions. In order to gauge the likelihood of CFD computational experiments to provide this benefit for analyses of buoyancy-induced flows in a BWR fuel assembly under complete LOCA conditions requires a suitable, simplified model in which comparable results are easily obtained using appropriate input values from the resulting CFD solutions. Unfortunately, consideration of existing program models intended for similar application (*i.e.* SFPT and GFLOW) reveal a number of inherent deficiencies (as previously discussed) that render the codes unsuitable for use in a comparative analysis.

The development of a buoyancy-induced flow program specific for use in the desired assessment is one of the objectives outlined in the current research. Initial consideration of preferred program attributes resulted in the additional constraint to implement a solution methodology that is also conducive to future subroutine appendages incorporating physical radiation and/or combined convection-conduction heat transfer mechanisms if desired. A precursory task is the identification of a well-validated analytical method used in similarly modeled systems to serve as a skeletal outline upon which specific programming code is constructed.

Extensive review of related topics in the literature reveals a strikingly analogous correlation between the physical phenomena associated with complete LOCA conditions and the normal operation environment in solar energy building technologies. The drive to reduce global energy consumption has increased the desire for new low-energy building solutions and awareness of the considerations involved in developing improved designs. As a result, emphasis on solar energy research realized the development of analytical methods and models for design analyses of related solar energy building

systems including: (1) Trombe wall solar heating systems (Smolec and Thomas, 1993; Uygur and Egrican, 1996); (2) porous wall-solar collector systems (Mbaye and Bilgen, 1992); (3) general configurations of obstructed channel systems with convection flows (Cruchaga and Celentano, 2003); (4) window enclosures with interior convective heat transfer (Muneer and Han, 1996); and (5) photovoltaic (PV) cladding ventilation systems (Yang *et al.*, 1996; Brinkworth *et al.*, 1997).

In particular, recent research efforts considering the effects of all hydraulic losses and developing inlet flow field profiles on design analyses of PV cooling ducts yielded a single loop solution method (Brinkworth *et al.*, 2000) and several subsequent model variations (Brinkworth, 2000; Brinkworth and Sandberg, 2005), explicitly derived for use in the estimation of buoyancy-induced flow rates in ventilated ducts with a net rate of internal heat gain. The simple, computationally efficient 'single loop' analysis with unique incorporation of a 'stratification parameter' for describing axial heat flux distribution (Brinkworth *et al.*, 2000), provides a sufficient analytical base method for development of a specific buoyant flow fuel assembly model applicable to complete LOCA scenarios. The exact tailoring of the single loop solution method required due to the diversity in modeled domain conditions and the final implementation into thermal hydraulic governing equations is discussed meticulously in Chapter 6 of this report. The research achievements noted above are intended to serve as the author's acknowledgment of the most important contributions afforded in the development of this program.

1.3 Related Modeling Efforts

Following several fundamental changes to MELCOR's original fuel model, including the implementation of new physical assumptions, the latest code development

(effectively termed the MELCOR SFP model) added the capability to model potential SFP accident domains. The first attempts to simulate a postulated SFP accident using this new model lead to the uncovering of a multitude of other considerations necessary in order to permit an accurate analysis of a typical SFP accident response. A concluding report stated that critical parameter values strongly affecting the SFP system behavior are unknown and other 'model calibration' work is needed (Gauntt and Wagner, 2004). The resolution of issues surrounding these critical modeling parameters, accuracy of implementation within designated MELCOR model packages, and a more robust understanding of buoyancy-driven flow in a complete loss-of-coolant accident are clear motivations behind Sandia National Laboratories' experimentally founded efforts to obtain basic thermal hydraulic data associated with modeling these incidents.

The overall approach by SNL to accomplish the stated objectives in a reasonable period requires a concurrent balance in research efforts involving aspects of basic analytical, experimental, and computational solution methods. This comprehensive consulting endeavor is divided among four smaller, more-manageable work propositions in which the overall objectives are met through a collaborative effort. Figure 1.2 provides a graphic overview of this project management narrative.

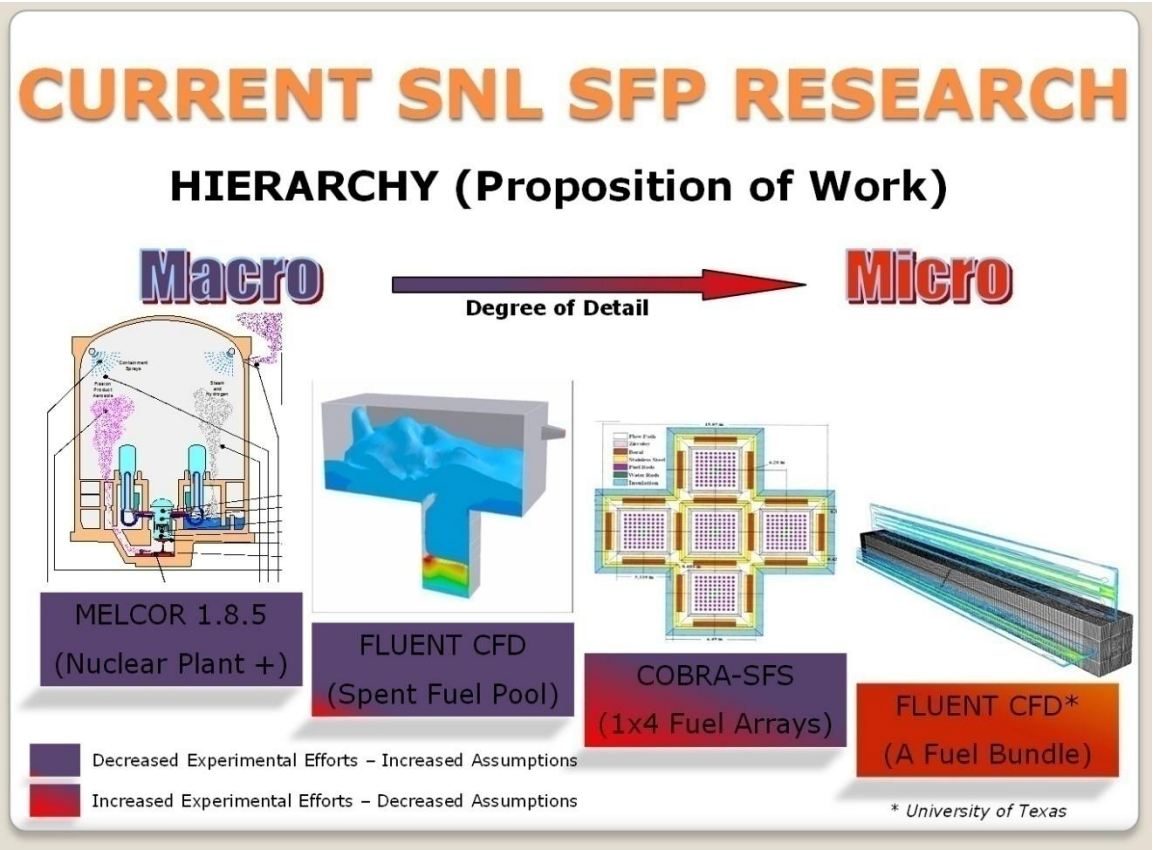


Figure 1.2: Different areas of related SFP research conducted at Sandia National Laboratories in response to NRC supported objectives.

The logic is rather simple and straightforward as experiments and resultant outcomes subject to less uncertainty have a higher probability of returning accurate data, solutions from these efforts are subsequently used as boundary conditions or parameter values (previously unknown) in more complex, larger domains to achieve well-founded outcomes in a continuing fashion (*i.e.* moving from right to left in Figure 1.2). This method or approach to solving quite large computation or constitutive subdomains of difficult to model physical circumstances is used to some extent by SNL in accomplishing the objectives set forth in related NRC supported research.

The experimental and analytical work of greatest relevance to the current research falls within confines of the microscopic degree of detail in Figure 1.2 representing 1x4 fuel array configuration studies with COBRA-SFS software, single fuel bundle assembly analyses utilizing commercial FLUENT CFD software, and the concurrent physical experiments performed at SNL (not shown) to which each is associated. These SNL experiments are derived with a single mindset to simulate thermal and hydraulic feedback of SFP assemblies under postulated complete loss-of-coolant conditions in a prototypic fashion. Objectives are directed towards collecting data on natural convection flow (simulated as forced flow with set rates similar in range to that expected by buoyancy-induced forces) in a single full length assembly, thermal radiation behavior within and among multiple assembly configurations, and the validation of measurements for use in calibrating the MELCOR 1.8.5 spent fuel pool model (Lindgren, 2004).

However, the extent of Sandia National Laboratories' research is not confined by these modeling concerns and facilitated code validation alone, work is undertaken in other macroscopic areas of proposition as well. In particular, the modeling of SFP building containment is of significant importance and closely correlated with the use of these experimental results (see Figure 1.3).

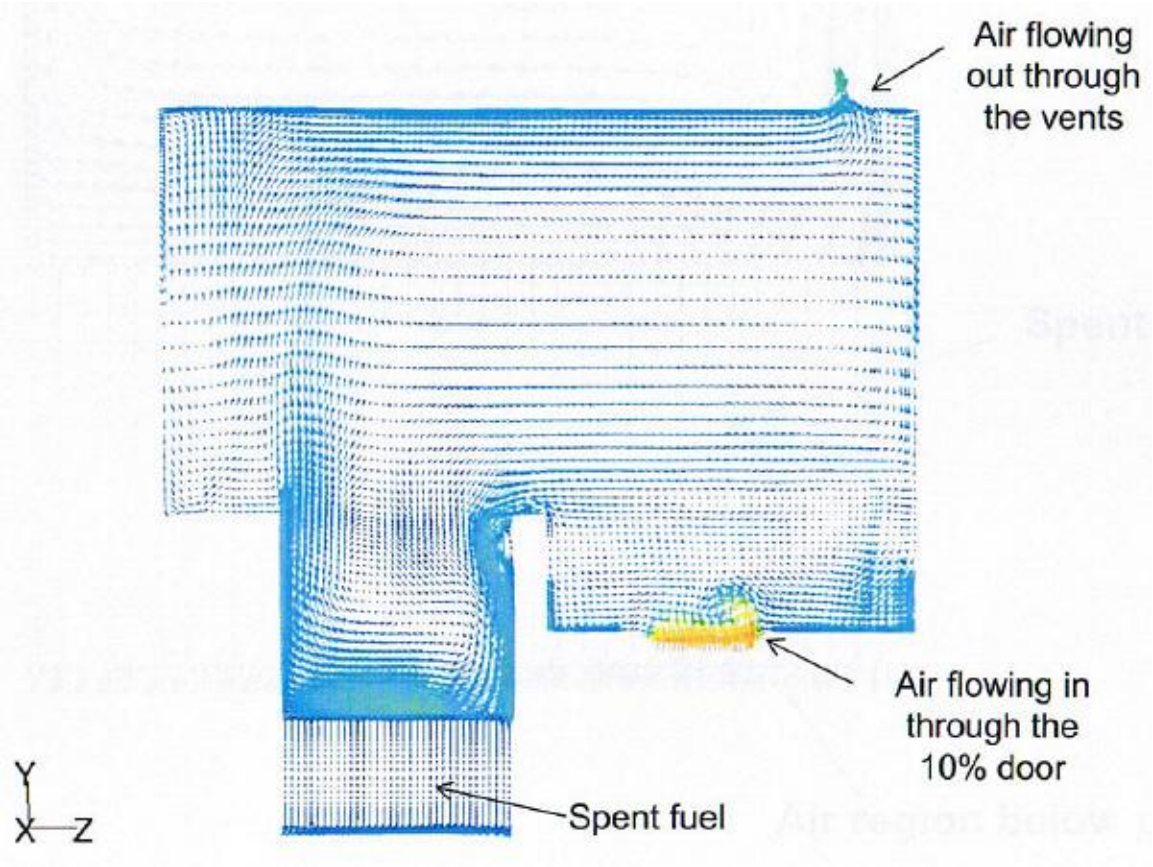


Figure 1.3: Airflow vectors in a SFP building from one of several postulated mitigation studies performed (Webb and Khalil, 2005).

Considerations stretch into mitigation strategies, SFP building airflow movements concerning proper ventilation procedures as well as venting equipment failures (Webb and Khalil, 2005), modeling test runs of newly calibrated MELCOR 1.8.5 SFP model code to assess improvements in accuracy, among several other issues. These are introduced in subsequent sections of this report if warranted and pertinent to the research subject discussed.

1.4 STATEMENT OF WORK

The impetus of this computational fluid dynamics modeling and numerical experimentation is to provide additional support and understanding of observed fluid flow behavior as well as the physical mechanisms that govern typical fuel assembly thermal hydraulic response in complete LOCA conditions. Commercial CFD software offers the capability to conduct numerical experiments to solve problems that are unfeasible in a normal laboratory setting and impart full flow-field solutions of the highest precision available among existing program codes with a similar field of application to the current research. This has spawned a significant interest at SNL to consider the potential results and benefits that computational modeling may afford through the utilization of commercial FLUENT CFD software in complete LOCA analyses of spent fuel assemblies. An extensive review of the literature re-emphasizes the superiority of this software, but also reveals the absence of any supporting attempts to utilize the commercial CFD software in a related manner to model an exact BWR fuel assembly geometry under the postulated accident conditions.

Therefore, this research is generally directed toward satisfying the need to provide a complete preliminary assessment of computational costs, benefits, model development considerations, solution uncertainties, simulation pitfalls, subsequent large-scale model obstacles, solution accuracy, and the potential application of results associated with the employment of FLUENT CFD software to analyze a BWR fuel assembly under complete LOCA conditions. Consideration of these aspects throughout the modeling development efforts and numerical simulations performed in each phase of this research also serves as an invaluable reference for guiding future modeling work and the expansion toward larger-scale simulations. The capabilities of the software and the accuracy of models

developed are discerned through comparisons with measured values determined by experiments conducted at Sandia National Laboratories.

Similarly, the CFD modeling efforts and computational experimental results provide supplementary high-fidelity solutions and additional characteristic hydraulic flow-field data supporting the calibration of accurate parameter values for the MELCOR SFP model in addition to the physical measurements obtained by SNL. The component of numerical CFD solution data gauging the viscous and form pressure losses is of significant importance, where a direct comparative assessment between major and minor hydraulic pressure loss coefficients from physical experiments at SNL is possible. Additional supporting evidence from computational simulations for the physical pressure loss experiments includes a combination of vital parameter values and 3D flow-field profiles for velocity, total pressure, and dynamic pressure across the assembly domain (in both forced and buoyancy-induced related flows), which are difficult measurements to obtain from the experimental setup but critical toward understanding/predicting the system response.

A number of principal tasks are undertaken throughout the course of this research with the purpose of aiding in the fulfillment of these objectives. In order to facilitate rapid and accurate construction of flow domains for mesh generation among CFD models created, a complete computer-aided design (CAD) solid model of Sandia National Laboratories' prototypic Global Nuclear Fuel (GNF) 9x9 BWR (GE 11 BWR/2-3) experimental fuel assembly is constructed using SolidWorks® 2005, commercial 3D CAD software (SolidWorks Corporation, 2005). The completed 3D solid 'assembly' file consists of mated solid 'part' files representing each individual component from which the entire fuel assembly is comprised (SolidWorks Corporation, 1995-2002).

Due to the proprietary nature of the fuel assembly design, a complete set of dimensioned drawings from the manufacturer is not attainable, although several completed component drawings obtained by SNL are available for reference. Therefore, remaining dimensions are individually measured in order to complete the first and only CAD solid model of the experimental assembly. This serves as an invaluable future reference and widely accessible resource to SNL researchers since final high-temperature experiments proposed result in the complete destruction of the prototypic experimental assembly (Lindgren, 2004).

As previously mentioned, a substantial amount of research effort is expended toward a sufficient preliminary assessment of considerations in modeling the domain and conditions of stated interest. Initial CFD models are setup to perform base case simulations and parameter studies of various mass flow rates within a theorized range of temperature induced buoyancy-driven airflow expected to develop through single-span fuel bundle segments. The analyses offer a detailed cross-examination for a multitude of resulting parametric values that are excellent for comparative evaluation. This includes preliminary scoping runs gauging resource requirements, grid-independence, periodicity and symmetry applicability, validity of viscous model options and wall treatment method (if one is needed), appropriateness of boundary conditions, and other pertinent setup characteristics which establish a sufficient level of accuracy and confidence in the final model implementation employed and solution(s) obtained.

Unfortunately, adding to the complexity of preliminary numerical simulations involving partial length segments of the fuel assembly is the lack of a sufficient inlet boundary condition profile, which only exists by explicit definition at the inlet nozzle of the lower tie plate (LTP). As a result, boundary condition profile studies are conducted utilizing short-length, fully-populated, single-span fuel bundle models in order to provide

some clarification on appropriate boundary conditions for extended domain models as well. Knowledge gained from the preliminary scoping simulations is incorporated into a subsequent larger domain model encompassing an entire fully-populated fuel bundle and grid spacer segment of single-span length from the prototypic experimental fuel assembly. Detailed flow-field and hydraulic pressure loss data from CFD simulations implementing this domain model are especially desirable to SNL due to complex gradients and high pressure losses expected with airflows traversing grid spacer components (personal communication with Eric Lindgren and Sam Durbin, February 28, 2008). Again, all of the computational experiments above are intended for comparative analysis of unheated, forced flow conditions analogous to physical pressure loss experiments conducted by SNL involving a prototypic experimental 9x9 BWR fuel assembly operated within the laminar flow regime; Reynolds numbers spanning 70 to 900 are characteristic of typical air mass flow rates examined.

Additional research focuses on the potential application of CFD simulation models representing the entire 9x9 BWR fuel assembly for use in both unheated, forced and heated, buoyancy-induced flow analyses. However, due to the requirement of further model development and investigational study regarding the use of high-fidelity, large production-type mesh models representing the entire assembly flow domain than achieved through the current stage of presented research, a simplified geometric domain model and approach is employed for initial evaluation of flow-field characteristics that are dependent upon the complete fuel assembly system. This is particularly important for predicting buoyancy-induced mass flow rates associated with varying levels of fuel decay heat input rate and the plausible development of alternative modeling approaches.

Computational experiments implementing a simplified geometric model of the entire BWR fuel assembly also afford supporting CFD solution data for validating the

development of a buoyancy-driven flow program using MATLAB® 6.5, commercial programming language and development platform software (The MathWorks, Inc., 1984-2002). This simplified program for predicting buoyancy-driven airflow rates within a prototypical BWR fuel assembly under complete LOCA conditions also functions as a complementary component to the aforementioned assessment of potential benefits offered by CFD simulation results through the provision of unknown critical parameter values governing the thermal hydraulic flow-field response. Again, the capability of acquiring accurate values for these parameters permits the formulation of a limited resource, computationally efficient model code that predicts reasonably accurate solutions to the same complex problem.

In summary, the primary objectives of this dissertation that offer new contributions to the field of open science research (to the best of the author's knowledge) are briefly recounted as follows:

1. Construction of the only detailed CAD solid model of Sandia National Laboratories' prototypic GNF 9x9 BWR experimental fuel assembly.
2. Development of the most high-fidelity geometric mesh model representations of complete BWR fuel assembly segments known in open science.
3. Execution of computational experiment simulations providing supplemental hydraulic flow field data supporting analogous pressure loss physical experiments conducted and measurements obtained at SNL.
4. Complete preliminary assessment of prevalent CFD modeling aspects in consideration of both the wide-ranging costs and benefits associated with the utilization of commercial FLUENT CFD software to analyze a BWR fuel

assembly under complete LOCA conditions and the implications of future, large-scale assembly mesh models characterizing the entire system.

5. Initial investigation of the potential application and benefits afforded by implementing CFD simulation models representing the complete 9x9 BWR fuel assembly for use in both unheated, forced and heated, buoyancy-induced flow analyses.
6. Development and evaluation of an original MATLAB program for predicting buoyancy-driven airflow rates within a prototypical BWR fuel assembly under complete LOCA conditions, which also functions as a complementary component to initially requisite evaluations of both unheated and heated flow-field solutions from simplified geometric model simulations of the entire BWR fuel assembly.

1.5 Consideration of Project Limitations

The use of powerful CFD modeling approaches and numerical techniques in high performance computing environments is by no means a standardized field in engineering research. In fact, the growth of a globally accepted set of policies and standards on which confidence in the accuracy and legitimacy of such numerical experimentation is reasonably established within the engineering community remain in an infancy stage of development at best. As a result, one of the biggest concerns in utilizing CFD models in modern day analyses, and where proponents of this new computational approach become particularly frustrated, is the frequent disregard for general engineering professionalism in the presentation of CFD related research. These considerations include efforts in which the numerical data is reasonably checked for accuracy, accompanied with an explanation of the basic governing physics modeled and numerical technique(s)

employed, and provided with sufficient clarity upon which informed readers may reproduce the results.

A strict guideline based on recommendations from several prestigious engineering journal publications (Roache *et al.*, 1986) is followed within reasonable efforts as introduced in Chapter 2 and discussed thoroughly in subsequent chapters where appropriate; assuring the validity, application, and accuracy of results with respect to the FLUENT CFD model used in the simulated response. This is particularly important in the current research due to the absence of any former, supporting attempts to model exact BWR fuel assembly geometry under the postulated accident conditions. Therefore, as previously mentioned, substantial effort is expended toward the assessment of fundamental considerations associated with the development of model geometry, meshing, simulation, postprocessing, and validation of results beginning with a phase of preliminary scoping studies. This is an integral practice in the appropriate progression of simulating and validating large production-type problems in order to identify numerical uncertainties and potential challenges in the initial model development stage. Solving these issues early is essential for minimizing the impact on computational cost and schedule (Hamman and Berry, 2009) as the resolution of unforeseen impediments is extremely problematic (if not impossible) on prohibitively large-scale models.

Unfortunately, similar modeling difficulties encountered in this research are also emphasized by other CFD practitioners attempting comparable large-scale simulations including large file sizes, extensive HPC resource requirements, significant geometry development and mesh generation times, and exorbitant postprocessing for data visualization (Hamman and Berry, 2009; Gajapathy *et al.*, 2007; Conner *et al.*, 2010). This is especially consistent with current simulations involving complex fuel components, which exhibit a wide range of geometric length scales across the flow

domain. It is reasonably understood that several modeling tradeoffs in the form of implied limitations toward consideration of the guideline steps are required in order to fulfill a more complete, meaningful scope of research.

Therefore, consideration of the impractical expenditure in effort required for complete model validation of the large-scale simulations precludes a substantial validation analysis outside of the models used in preliminary scoping studies. The results of these initial case studies and model validation assessments are assumed to accurately reflect the greater fuel assembly domain and effectively achieve the original intentions of the research objectives outlined. Similar margins are imposed in the postprocessing of graphical results associated with the large-scale simulations. Although the modeling tradeoffs are undesirable, the imposed limitations are congruent with the progressive approach methodology of simulating and validating large production-type problems beginning with a phase of preliminary scoping simulations over small segments of the entire system, as similarly suggested in related CFD research efforts (Hamman and Berry, 2009).

Additional project limitations of noteworthy significance are inherent to the uncontrollable circumstances associated with the segregation of computational and physical experiments conducted in the current research and at SNL, respectively. These limits are encompassed by uncertainties in the accuracy of hand measurements for absent component dimensions, unknown variances between available commercial component drawings and actual mock components used in the physical experiment apparatus, and several post-computational model development modifications to the SNL experimental setup. Other constraints are imposed by span lengths between and locations of pressure measurement ports in the physical experiment not conducive to the designation of appropriate CFD model inlet/outlet boundary conditions as well as considerable

variations in ambient conditions between subsequent physical pressure loss measurements across segments of the fuel assembly, which are taken over a span of several days. The accuracy of a direct comparative analysis between computational and physical experiments is certainly hindered by these circumstances, but absence of an appropriate gauge for these inaccuracies requires the assumption of negligible contribution at this stage of presented research.

Little guidance is available for approximating computer resource/hardware requirements, sufficient CPU-hour dedications, or even the time investment required for model geometry construction and meshing in large-scale simulations, which often add to the project limitations already identified. Careful consideration of these modeling aspects is extremely important when outlining a systematic approach that is both cost effective and time efficient with the least associated risk-of-failure. Despite extensive effort to alleviate the number of research limitations in desired CFD modeling analyses, such attempt to eliminate every margin is futile; therefore, pertinent constraints coupled with computational analyses are further detailed in appropriate chapters throughout this dissertation. The aforementioned project limitations are constituent factors considered in the ultimate approach of all modeling efforts undertaken with regard to this research. As a result, this dissertation represents the author's best effort to impart a complete preliminary assessment of employed commercial FLUENT CFD software to model the domain and conditions of stated interest as well as provide the most beneficial, accurate numerical solution data in support of Sandia National Laboratories' research objectives.

Chapter 2: CFD Numerical Experiment Framework

2.1 Relevant Theory

Aside from the obvious preprocessor, solver, and postprocessor functionality, a number of novice commercial CFD software users fail to possess a clear understanding or knowledge of the governing equations and mathematical algorithms used to model physical mechanisms within a system domain and resolve the flow field associated with a particular problem. The CFD software is effectively treated as a black box entity with a desired function but whose constituent numerical methods are unknown, which inherently constitutes a very risky practice. In order to briefly address these concerns and provide appeal to readers over a wide range of modeling experience, the fundamental theory incorporated throughout this research is discussed in the following chapter sections that is essential for a general understanding of computational experiments and solutions presented in subsequent chapters. This includes adequate review of basic governing flow equations, transport theory, SIMPLE solution algorithm, turbulent flow, and buoyancy force, which introduces relevant flow model developments in CFD software through the application of physical principles. Substantial discussion is devoted toward formulation, discretization, and implementation of the Navier-Stokes equations in modeling basic fluid flows as well as finite-volume numerical solution techniques.

However, a few noteworthy comments regarding the intent and approach of this effort are warranted due to expansiveness of the subjects mentioned. There are numerous approaches for deriving any of the basic governing equations of fluid flow with each approach pertaining to a different model representation of the flow field and a specific resulting form of the equation(s) considered. Dependence of the evaluation method to

characterize a system in the flow as either a finite control volume or an infinitesimally small element, determines whether the equation is obtained directly in integral or partial differential form, respectively. Furthermore, the system representation in the method employed is characterized as fixed in space (*i.e.* an Eulerian representation) or of fixed mass moving with the flow (*i.e.* a Lagrangian representation), which leads to the additional specification of either a conservation or non-conservation form of the equation, respectively. The associated forms of the governing equations derived from application of these different system models of the flow are illustrated in Figure 2.1 below using the continuity equation as an example.

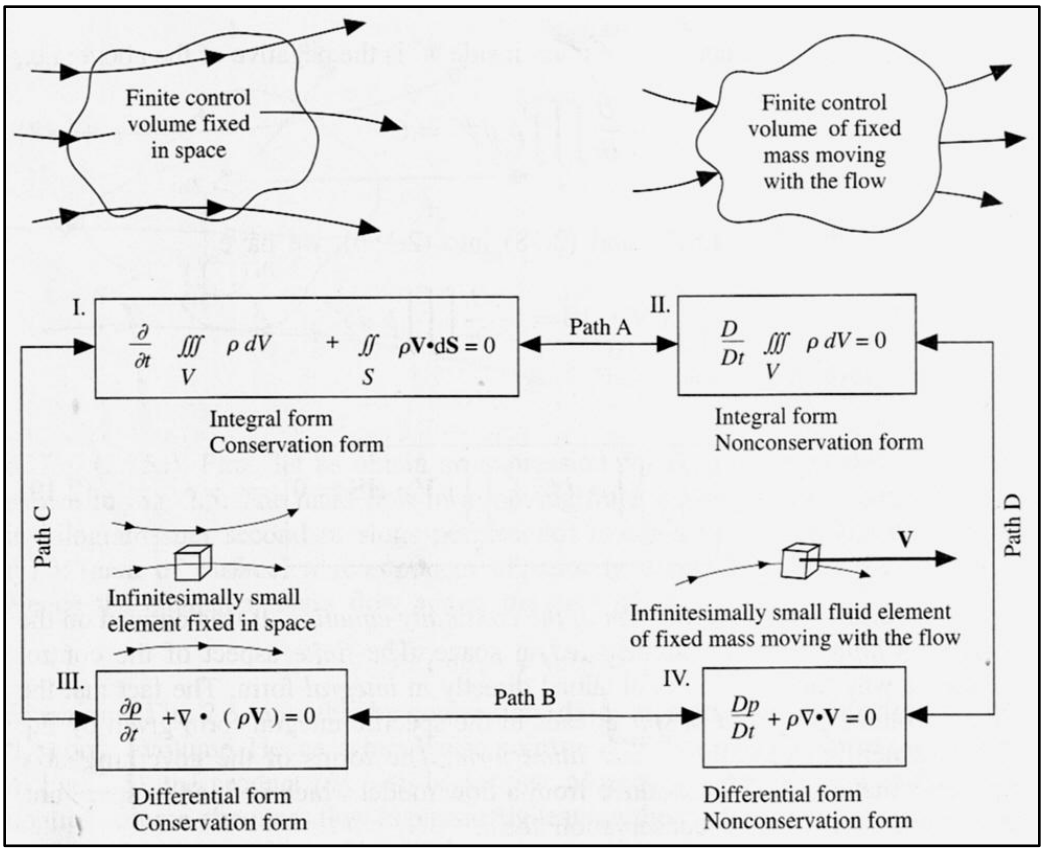


Figure 2.1: Derived forms of the continuity equation resulting from different system models of the flow (Anderson, 1995).

Although Figure 2.1 only emphasizes the relationship between different system models and forms of the continuity equation, the same philosophy and methods apply in the formulation of the momentum and energy equations as well. The initial purpose of introducing these development facets of the basic equations of fluid motion is simply to demonstrate an acknowledgment of the numerous equation forms and diverse approaches of derivation embodying the subject. As a result, this achieves the foremost intent to accentuate the significance of the following imperative, underlying aspects in Figure 2.1 concerning extraction of the four different mathematical statements: (1) the four different equations are not fundamentally disparate equations at all, but rather four different forms of the same equation and (2) any of the four equation forms can be derived by manipulation from any of the others (Anderson, 1995).

In other words, any model of the flow and resulting form of the governing equation(s) is an equivalent statement of the same fundamental physical principles as the other three alternative representations; therefore, preference is specific to application and/or ease of derivation (to the extent of concern in this research). Furthermore, it is sufficient to derive the governing equation(s) utilizing any model of the flow and subsequently obtain a separate, desired form of the equation(s) by manipulation of the current form using generally simple transforming relations. The implications are eminent as rigorous mathematical derivation of each governing flow equation in all possible forms is undoubtedly beyond the scope of consideration in the following sections; however, the prevailing statements above advocate the superfluousness of such an endeavor.

Based on these observations, a suitable approach is outlined for the following discussions with the intent to impart a concise, perspicuous delineation of the most-relevant aspects regarding the governing flow equations and numerical methods integrated into the CFD segments of this research. The continuity and momentum

equation derivations are achieved by exploiting infinitesimally small element models of the flow due to the straightforward representation of the physical conservation principles in the model diagrams. Explicitly, an Eulerian description of the flow is apt for mass flow rate balance across element surfaces and a Lagrangian description is particularly convenient in conjunction with Newton's second law, owing to the element's fixed mass moving with the flow. Therefore, the continuity equation is obtained in partial differential-conservation equation form and the momentum equation is obtained in partial differential-non-conservation equation form. The resulting complete Navier-Stokes equations are transformed to attain a partial differential-conservation equation form to demonstrate the indirect manipulation of one equation form to another, as well as present a common form of the Navier-Stokes equations found in fluid mechanics literature.

The differential and integral forms of general transport equations employed often throughout CFD algorithms arise here from an observation of the similarities between the governing fluid flow equations. In the development of these general transport equations, discussed further in Section 2.1.2, transformation from the differential to integral equation form used in the finite-volume method is illustrated. This integral-conservation equation form is the most common form of the governing flow equations utilized in commercial CFD software, including FLUENT. The discretization of this general equation form is introduced in Section 2.1.2 and then briefly addressed in Section 2.2 with specific regard to the computational solver in the FLUENT CFD software.

On a final note, detailed formulation of the energy equation is precluded by similarity of the derivation with the momentum equation as well as the limited application of the basic equation form in a few CFD models implemented in this research. In addition, it is assumed that nomenclature used in presenting the general governing flow equations is commonplace enough throughout introductory fluid

mechanics and CFD literature that readers possessing a basic familiarity of the subject do not warrant comprehensive definition of included parameters and notation. However, references to material concerning fundamental fluid mechanics and transport phenomenon (Alexandrou, 2001; Fox and McDonald, 1985, 1992; Young *et al.*, 2004; Bird *et al.*, 1960), equation derivations specific to CFD numerical algorithms (Anderson, 1995; Versteeg and Malalasekera, 1995), and common mathematical notations (Spiegel and Liu, 1999) are noted if additional information concerning the following discussions is required.

2.1.1 Governing Equations

The governing equations in fluid dynamics are simply just avowals of widely known conservation laws of physics in a mathematical set of functions embracing the conservation of mass, momentum, and energy. The initial steps in deriving any of these fluid flow equations include the selection of an appropriate physical principle, application of the principle to a representative model of the flow and balance diagram, and construction of a statement expressing the principle as applied to the model of the flow. In consideration of the continuity equation, the physical principle of concern is the conservation of mass and a suitable model of the flow is an infinitesimally small element fixed in space. This model of the flow and an associated diagram of mass flow through the fixed element are illustrated in Figure 2.2 below.

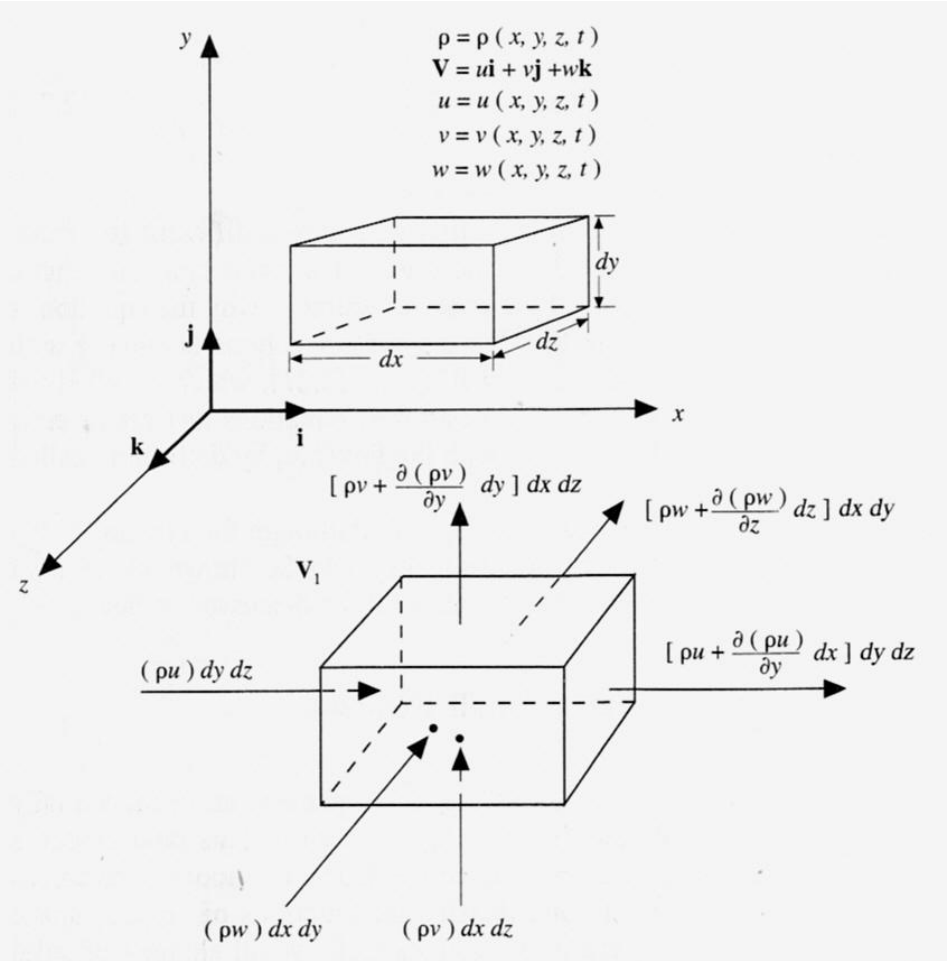


Figure 2.2: Model of the flow and diagram of mass fluxes through the element surfaces for derivation of the continuity equation (Anderson, 1995).

The conservation of mass, as applied to the fixed element in Figure 2.2, is then appropriately expressed in a statement of the conservation principle as follows:

$$Time\ rate\ of\ mass\ increase\ in\ element = Net\ mass\ flow\ into\ element$$

Focusing on the right-hand side of this statement and denoting an inflow of mass as a positive quantity in Figure 2.2, the net inflow of mass across the element boundaries for each direction are written as (correcting the typographical error in Anderson's diagram)

$$Net\ inflow\ in\ x\_dir = (\rho u)\,dy\,dz - \left[\rho u + \frac{\partial(\rho u)}{\partial x}dx\right]dy\,dz = -\frac{\partial(\rho u)}{\partial x}dx\,dy\,dz$$

$$Net\ inflow\ in\ y\_dir = (\rho v)\,dx\,dz - \left[\rho v + \frac{\partial(\rho v)}{\partial y}dy\right]dx\,dz = -\frac{\partial(\rho v)}{\partial y}dx\,dy\,dz$$

$$Net\ inflow\ in\ z\_dir = (\rho w)\,dx\,dy - \left[\rho w + \frac{\partial(\rho w)}{\partial z}dz\right]dx\,dy = -\frac{\partial(\rho w)}{\partial z}dx\,dy\,dz$$

Hence, the net mass flow into the element is

$$Net\ mass\ flow\ into\ element = -\left[\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z}\right]dx\,dy\,dz \qquad (2.1)$$

Turning attention toward the left-hand side of the conservation statement and noting the total mass of fluid in the infinitesimally small element is simply given by the product of density and element volume, the time rate of mass increase inside the element is

$$Time\ rate\ of\ mass\ increase = \frac{\partial \rho}{\partial t}dx\,dy\,dz \qquad (2.2)$$

Thus, the conservation statement in terms of Equations (2.1) and (2.2) is expressed as

$$\frac{\partial \rho}{\partial t} dx\, dy\, dz = -\left[\frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z}\right] dx\, dy\, dz \tag{2.3}$$

Upon simplifying, this becomes

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0 \tag{2.4}$$

or, in compact vector notation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = \frac{\partial \rho}{\partial t} + div(\rho \boldsymbol{V}) = 0 \tag{2.5}$$

Equations (2.4) and (2.5) are equivalent representations of the unsteady, compressible continuity equation obtained in partial differential-conservation form.

If the fluid is incompressible, the density is constant; it is neither a function of space nor time. Therefore, Equation (2.4) becomes

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \tag{2.6}$$

again, in compact vector notation

$$\nabla \cdot \boldsymbol{V} = div(\boldsymbol{V}) = 0 \tag{2.7}$$

Equations (2.6) and (2.7) are equivalent representations of the continuity equation for an incompressible fluid flow. The steady-state flow constraint, by definition, asserts that all fluid properties are independent of time, which is intrinsic to the incompressibility

condition in the continuity equations above. Furthermore, it is important to clarify the occasional misconception pertaining to the conventional terminology used in labeling the continuity equation forms above. Contrary to the designation as unsteady, compressible representations of the continuity equation, Equations (2.4) and (2.5) are valid for steady or unsteady flow, and compressible or incompressible fluids. The specific labeling ascribed is actually intended as a statement of additional constitutive relations included in the derivation beyond the most basic requirements of an incompressible fluid flow. Hence, the incompressible forms of the continuity equation given by Equations (2.6) and (2.7) are not applicable for steady or unsteady flow of compressible fluids.

Derivation of the momentum governing flow equation follows a similar approach and the fundamental principle of concern is the conservation of momentum, namely Newton's second law. An infinitesimally small fluid element of fixed mass moving with the flow is elected as the preferred model of the flow and surface forces acting on the element in the x-direction are shown in Figure 2.3 for reference.

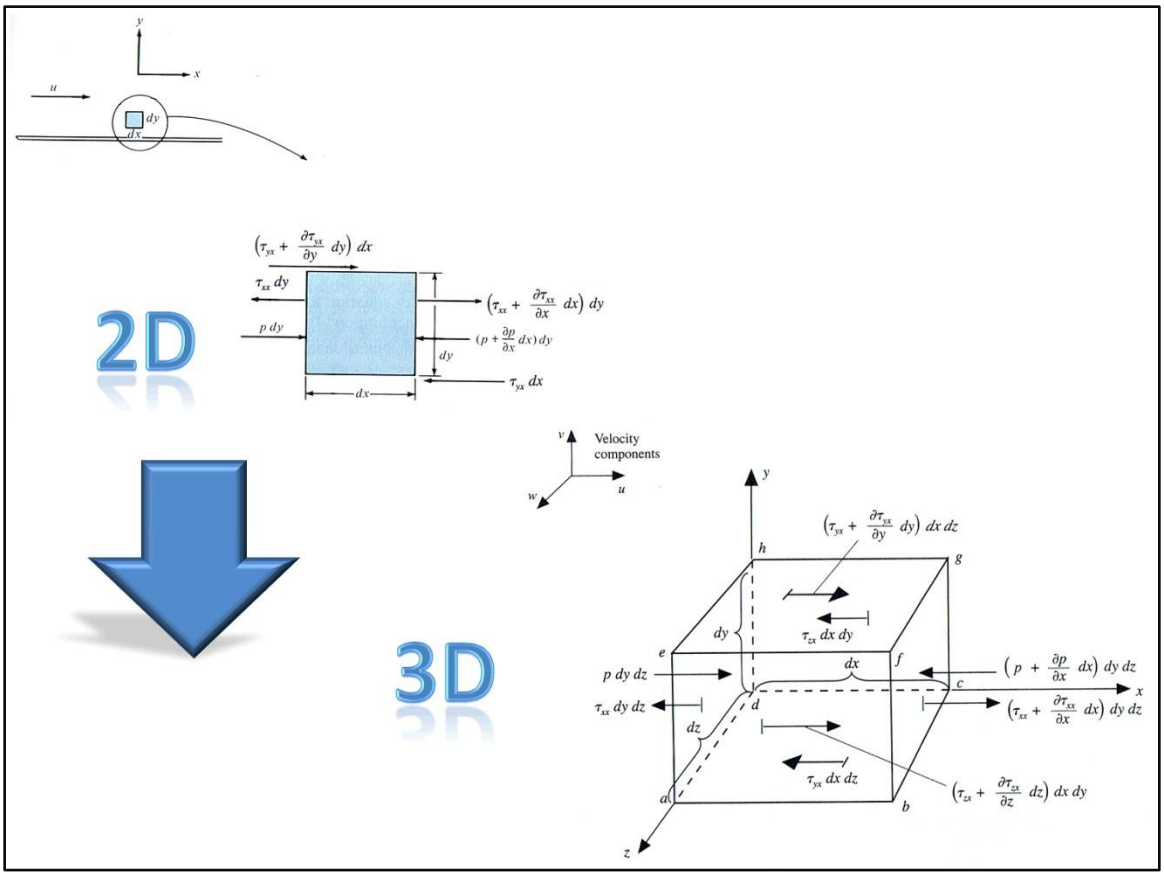


Figure 2.3: Surface forces acting on fluid element in x-direction used in deriving the x-component of the momentum equation; adapted in part from (Holman, 1990; Anderson, 1995).

Newton's second law, as applied to the designated model of the flow, is conveyed in a statement of the conservation principle as follows:

$$Rate\ of\ momentum\ increase\ of\ fluid\ element = Net\ force\ on\ fluid\ element$$

This is a vector relation consisting of three scalar components. Therefore, considering only the x-component of this statement initially, the representative equation becomes

$$ma_x = F_x \tag{2.8}$$

Equation (2.8) is a fundamental avowal of the x-component of Newton's second law and the focal point of derivation for the x-component of the momentum equation.

The term on the right-hand side of Equation (2.8) constitutes body forces as well as surface forces. The body forces act directly on the volumetric mass of the fluid element, where the mass of the fluid element is simply a product of the element density and volume. Therefore, if the x-component of a generalized body force per unit mass is assigned the term (f_x), the body force acting on the fluid element in the x-direction is given by

$$Body\ force\ acting\ on\ fluid\ element\ in\ x\_dir = \rho f_x\, dx\, dy\, dz$$

Surface forces, which act directly on the surface of the fluid element, are a result of pressure, shear, and normal stress distributions imposed by the surrounding outer fluid. Using the surface forces diagram in Figure 2.3 for the x-direction and delegating a positive sign convention for forces in the positive direction of a coordinate axis (viscous force directions are designated at surfaces based on increasing components of velocity in the positive directions of the axes), the total fluid element surface force in the x-direction is written as

$$Total\ surface\ force\ in\ x\_dir = \left[-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right] dx\, dy\, dz$$

Hence, combining expressions for the total surface and body forces acting on a fluid element in the x-direction, the net force on the fluid element represented by the (F_x) term on the right-hand side of Equation (2.8) is

$$F_x = \left[-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right] dx\, dy\, dz + \rho f_x\, dx\, dy\, dz \tag{2.9}$$

The left-hand side of Equation (2.8) is the rate of x-momentum increase of the fluid element, where the term for mass of the fluid element is fixed for this model of the flow and equal to

$$m = \rho\, dx\, dy\, dz \tag{2.10}$$

In order to define the x-component of acceleration, a physical understanding and numerical interpretation of the velocity field traversed by the fluid element is necessary. Following the movement of the fluid element within the velocity field, two potential factors influence a change in velocity: (1) the local flow-field velocity itself is fluctuating with time and (2) the fluid element is moving to another location in the flow field where the velocity is spatially different. Therefore, the velocity of the fluid element is a function of both location and time with the x-component of the velocity field given by

$$u = u(x, y, z, t)$$

Using the chain rule/definition of a multivariable differential yields

$$du = \frac{\partial u}{\partial x}dx + \frac{\partial u}{\partial y}dy + \frac{\partial u}{\partial z}dz + \frac{\partial u}{\partial t}dt$$

and rearranging,

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + \frac{\partial u}{\partial x}\frac{dx}{dt} + \frac{\partial u}{\partial y}\frac{dy}{dt} + \frac{\partial u}{\partial z}\frac{dz}{dt}$$

Noting that the fluid element follows the flow in the described model of the flow selected for this derivation, known relations for the velocity are

$$\frac{dx}{dt} = u, \qquad \frac{dy}{dt} = v, \qquad \frac{dz}{dt} = w$$

Substituting these relations into the above statement gives

$$\frac{du}{dt} = \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}$$

Therefore, since the time rate of change of the x-component of velocity defines the x-component of acceleration for the fluid element, the concluding expression is

$$\frac{Du}{Dt} \equiv a_x = \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} \tag{2.11}$$

The derivative term in Equation (2.11) is given a unique symbol (*D/Dt*) since the derivative represents a fluid element of specific substance or mass. This relationship holds for any flow-field variable (not just velocity), and is commonly referred to as the

'substantial derivative' (Young *et al.*, 2004). Physically, the substantial derivative is the time rate of change of a flow-field variable following a moving fluid element of specific substance. A general definition for this special derivative is given by the following delineation

$$\frac{D(\,)}{Dt} \equiv \frac{\partial(\,)}{\partial t} + u\frac{\partial(\,)}{\partial x} + v\frac{\partial(\,)}{\partial y} + w\frac{\partial(\,)}{\partial z} \tag{2.12}$$

or, in vector notation

$$\frac{D(\,)}{Dt} \equiv \frac{\partial(\,)}{\partial t} + (\boldsymbol{V}\cdot\nabla)(\,) \tag{2.13}$$

As a result, Equations (2.10) and (2.11) combine to give the rate of x-momentum increase of the fluid element, which is represented by the terms on the left-hand side of Equation (2.8) as

$$ma_x = \rho\, dx\, dy\, dz\frac{Du}{Dt} \tag{2.14}$$

Thus, substituting Equations (2.9) and (2.14) into Equation (2.8), the x-component of the momentum equation becomes

$$\rho\, dx\, dy\, dz\frac{Du}{Dt} = \left[-\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right] dx\, dy\, dz + \rho f_x\, dx\, dy\, dz \tag{2.15}$$

Dividing through by the volume of the fluid element gives

$$\rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho f_x \tag{2.16}$$

The x-component of the momentum equation following a moving fluid element within the flow field is given by Equation (2.16). The same procedure is applied for deriving the y- and z-components of the momentum equation as outlined above for the x-direction. These equations are written as

$$\rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho f_y \tag{2.17}$$

$$\rho \frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho f_z \tag{2.18}$$

Equations (2.16), (2.17), and (2.18) represent the x-, y-, and z-components, respectively, of the unsteady, compressible momentum equation obtained in partial differential-non-conservation form.

The partial differential-conservation form of these equations are obtainable through the use of a simple transform relation, which is derived by writing out the left-hand side in terms of the substantial derivative definition, applying the vector identity for the divergence of a scalar and vector product, and substituting for the collection of terms representing the continuity equation [Equation (2.5)]. A detailed formulation is not shown here for reasons of brevity; however, the resulting expression for the left-hand side of the x-component momentum equation [Equation (2.16)] in vector notation is (Anderson, 1995)

$$\rho \frac{Du}{Dt} = \frac{\partial (\rho u)}{\partial t} + \nabla \cdot (\rho u \boldsymbol{V}) \tag{2.19}$$

Using the transform relation given by Equation (2.19), the x-component of the momentum equation [Equation (2.16)] is equivalently expressed in vector notation as

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u \boldsymbol{V}) = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho f_x \tag{2.20}$$

Similarly, by implementing an identical vector form of the transform relation for the y- and z-component directions, Equations (2.17) and (2.18) become

$$\frac{\partial(\rho v)}{\partial t} + \nabla \cdot (\rho v \boldsymbol{V}) = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho f_y \tag{2.21}$$

$$\frac{\partial(\rho w)}{\partial t} + \nabla \cdot (\rho w \boldsymbol{V}) = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho f_z \tag{2.22}$$

Equations (2.20), (2.21), and (2.22) represent the x-, y-, and z-components, respectively, of the unsteady, compressible momentum equation written in compact vector notation and obtained in partial differential-conservation form. Again, it is important to emphasize the equivalence of both forms of the momentum equation expressed in Equations (2.16) through (2.18) and Equations (2.20) through (2.22), and note that both forms always have an identical collection of terms on the right-hand side of the equality despite utilizing a different model of the flow to arrive at each equation form.

Unfortunately, in order to solve either form of the momentum equation stated above, suitable expressions relating the stresses in terms of the velocity gradients and fluid properties are required. Otherwise, there are more unknowns than equations available for resolving the flow field. Newton's law of viscosity states that the rate of

deformation for certain fluids is proportional to the local viscous stress state of the fluid. Fluids characterized by this constitutive relation (*i.e.* stress state is proportional to the velocity gradients in the fluid) are commonly referred to as Newtonian fluids. In 1845, Stokes obtained the following expressions for the viscous stresses in Newtonian fluids (Anderson, 1995):

$$\tau_{xx} = \lambda(\nabla \cdot \boldsymbol{V}) + 2\mu\frac{\partial u}{\partial x} \tag{2.23a}$$

$$\tau_{yy} = \lambda(\nabla \cdot \boldsymbol{V}) + 2\mu\frac{\partial v}{\partial y} \tag{2.23b}$$

$$\tau_{zz} = \lambda(\nabla \cdot \boldsymbol{V}) + 2\mu\frac{\partial w}{\partial z} \tag{2.23c}$$

$$\tau_{xy} = \tau_{yx} = \mu\left[\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right] \tag{2.23d}$$

$$\tau_{xz} = \tau_{zx} = \mu\left[\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right] \tag{2.23e}$$

$$\tau_{yz} = \tau_{zy} = \mu\left[\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right] \tag{2.23f}$$

Expanding the vector notation on the left-hand side of Equations (2.20), (2.21), and (2.22), followed by the substitution of Equations (2.23) for the viscous stress terms, the x-, y-, and z-components of the momentum equation are written respectively as

$$\frac{\partial(\rho u)}{\partial t}+\frac{\partial(\rho u^2)}{\partial x}+\frac{\partial(\rho uv)}{\partial y}+\frac{\partial(\rho uw)}{\partial z}=-\frac{\partial p}{\partial x} \\ +\frac{\partial}{\partial x}\left(\lambda\nabla\cdot\boldsymbol{V}+2\mu\frac{\partial u}{\partial x}\right)+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]+\rho f_x \tag{2.24}$$

$$\frac{\partial(\rho v)}{\partial t}+\frac{\partial(\rho uv)}{\partial x}+\frac{\partial(\rho v^2)}{\partial y}+\frac{\partial(\rho vw)}{\partial z}=-\frac{\partial p}{\partial y} \\ +\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\right)\right]+\frac{\partial}{\partial y}\left(\lambda\nabla\cdot\boldsymbol{V}+2\mu\frac{\partial v}{\partial y}\right)+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)\right]+\rho f_y \tag{2.25}$$

$$\frac{\partial(\rho w)}{\partial t}+\frac{\partial(\rho uw)}{\partial x}+\frac{\partial(\rho vw)}{\partial y}+\frac{\partial(\rho w^2)}{\partial z}=-\frac{\partial p}{\partial z} \\ +\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)\right]+\frac{\partial}{\partial z}\left(\lambda\nabla\cdot\boldsymbol{V}+2\mu\frac{\partial w}{\partial z}\right)+\rho f_z \tag{2.26}$$

Equations (2.24), (2.25), and (2.26) represent the complete Navier-Stokes equations for unsteady, compressible fluid flow in partial differential-conservation form.

In an effort to illustrate the significant commonalities between the governing flow equations, Versteeg and Malalasekera (1995) have suggested recasting the Navier-Stokes equations above in a slightly different manner. The authors achieve a simplified set of the equations by separating the terms representing viscous resistance to expansion/compression of a gas (*i.e.* $\lambda\nabla\cdot\boldsymbol{V}$) from the other viscous stress terms and combine them with the body forces in a single momentum source term (S_M). All remaining viscous stresses are rearranged into a solitary group of terms that is easily expressed in compact vector notation. The resulting Navier-Stokes equations as well as continuity, energy, and state equations for an unsteady, compressible fluid flow in partial differential-conservation form are included in Figure 2.4 below.

Mass	$\frac{\partial \rho}{\partial t} + div(\rho \mathbf{u}) = 0$
x-momentum	$\frac{\partial (\rho u)}{\partial t} + div(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + div(\mu \, grad \, u) + S_{Mx}$
y-momentum	$\frac{\partial (\rho v)}{\partial t} + div(\rho v \mathbf{u}) = -\frac{\partial p}{\partial y} + div(\mu \, grad \, v) + S_{My}$
z-momentum	$\frac{\partial (\rho w)}{\partial t} + div(\rho w \mathbf{u}) = -\frac{\partial p}{\partial z} + div(\mu \, grad \, w) + S_{Mz}$
Internal energy	$\frac{\partial (\rho i)}{\partial t} + div(\rho i \mathbf{u}) = -p \, div \, \mathbf{u} + div(k \, grad \, T) + \Phi + S_i$
Equations of state	$p = p(\rho, T)$ and $i = i(\rho, T)$ e.g. perfect gas $p = \rho RT$ and $i = C_v T$

Figure 2.4: Governing flow equations for an unsteady, compressible fluid flow of a Newtonian fluid (Versteeg and Malalasekera, 1995).

On a final note, there are a number of situations where the fluid properties and associated flow-field conditions under evaluation are reasonably characterized by incompressible, constant viscosity flow assumptions. Under these conditions, the complete Navier-Stokes equations for an unsteady, compressible flow [Equations (2.24), (2.25), and (2.26)] are greatly simplified. The basic embodiments of the Navier-Stokes equations for incompressible, constant viscosity flow are directly obtainable from the corresponding x-, y-, and z-component equations for unsteady, compressible flow specified by Equations (2.24), (2.25), and (2.26), respectively.

As an example of the formulation development, consider the Navier-Stokes equation in the x-direction for unsteady, compressible fluid flow [Equation (2.24)]. Focusing on the right-hand side of Equation (2.24), the terms explicitly involving the divergence of the velocity vector cancel out as defined by the incompressible continuity equation [Equation (2.7)] and viscosity terms are moved outside the partial derivatives

since the viscosity is held constant. Implementing these revisions, Equation (2.24) becomes

$$\frac{\partial(\rho u)}{\partial t}+\frac{\partial(\rho u^2)}{\partial x}+\frac{\partial(\rho uv)}{\partial y}+\frac{\partial(\rho uw)}{\partial z}=-\frac{\partial p}{\partial x} \\ +2\mu\frac{\partial^2 u}{\partial x^2}+\mu\frac{\partial^2 v}{\partial x\partial y}+\mu\frac{\partial^2 u}{\partial y^2}+\mu\frac{\partial^2 u}{\partial z^2}+\mu\frac{\partial^2 w}{\partial x\partial z}+\rho f_x \tag{2.27}$$

The right-hand side of Equation (2.27) is further reduced by obtaining an expression for the second term to the right of the equality using the continuity equation in the form given by Equation (2.6), which is rearranged for the partial derivative of the x-component of velocity as

$$\frac{\partial u}{\partial x}=-\frac{\partial v}{\partial y}-\frac{\partial w}{\partial z} \tag{2.28}$$

Differentiating Equation (2.28) with respect to the x-direction gives

$$\frac{\partial^2 u}{\partial x^2}=-\frac{\partial^2 v}{\partial x\partial y}-\frac{\partial^2 w}{\partial x\partial z} \tag{2.29}$$

Adding the second partial derivative of the x-component of velocity to both sides of Equation (2.29) and multiplying through by the viscosity yields

$$2\mu\frac{\partial^2 u}{\partial x^2}=\mu\frac{\partial^2 u}{\partial x^2}-\mu\frac{\partial^2 v}{\partial x\partial y}-\mu\frac{\partial^2 w}{\partial x\partial z} \tag{2.30}$$

Substituting Equation (2.30) into Equation (2.27) for the second term to the right of the equality and canceling similar terms, Equation (2.27) becomes

$$\frac{\partial(\rho u)}{\partial t}+\frac{\partial(\rho u^2)}{\partial x}+\frac{\partial(\rho uv)}{\partial y}+\frac{\partial(\rho uw)}{\partial z}=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\rho f_x \quad (2.31)$$

Turning attention toward the left-hand side of Equation (2.31), there are a few additional simplifications, which are practicable under the assumptions of incompressible, constant viscosity fluid flow. Partially expanding the derivatives on the left-hand side of Equation (2.31) gives

$$\begin{aligned}\rho\frac{\partial u}{\partial t}+u\frac{\partial \rho}{\partial t}+\rho u\frac{\partial u}{\partial x}+u\frac{\partial(\rho u)}{\partial x}+\rho v\frac{\partial u}{\partial y}+u\frac{\partial(\rho v)}{\partial y} \\ +\rho w\frac{\partial u}{\partial z}+u\frac{\partial(\rho w)}{\partial z}=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\rho f_x\end{aligned} \quad (2.32)$$

Rearranging and grouping meaningful terms together, Equation (2.32) becomes

$$\begin{aligned}\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)+u\left[\frac{\partial \rho}{\partial t}+\frac{\partial(\rho u)}{\partial x}+\frac{\partial(\rho v)}{\partial y}+\frac{\partial(\rho w)}{\partial z}\right] \\ =-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\rho f_x\end{aligned} \quad (2.33)$$

The bracketed terms on the left-hand side of Equation (2.33) are an exact statement of the continuity equation defined by Equation (2.4); hence, the bracketed terms are equal to zero. Therefore, Equation (2.33) is written in final simplified form as

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=-\frac{\partial p}{\partial x}+\mu\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+\rho f_x \qquad (2.34)$$

Following a similar procedure, the y- and z-component equations are derived from corresponding unsteady, compressible Equations (2.25) and (2.26), yielding

$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}\right)=-\frac{\partial p}{\partial y}+\mu\left(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right)+\rho f_y \qquad (2.35)$$

$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=-\frac{\partial p}{\partial z}+\mu\left(\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right)+\rho f_z \qquad (2.36)$$

Equations (2.34), (2.35), and (2.36) are the x-, y-, and z-components, respectively, which represent the complete Navier-Stokes equations for an incompressible, constant viscosity fluid flow obtained directly through simplification of the unsteady, compressible equivalent form of the Navier-Stokes equations.

2.1.2 Transport Theory and SIMPLE Solution Algorithm

The differential and integral forms of general transport equations employed often throughout CFD algorithms arise here from an observation of the similarities between the governing equations of fluid flow derived above, particularly the form of these equations presented in Figure 2.4. Defining any general variable quantity or property scalar as (ξ) and hiding any unshared terms between the equations in the source terms, it is evidently clear that a basic transport equation for (ξ) in partial differential-conservation form is written as follows:

$$\frac{\partial(\rho\xi)}{\partial t} + div(\rho\xi \boldsymbol{u}) = div(\Gamma\ grad\ \xi) + S_\xi \tag{2.37}$$

In words, the terms in Equation (2.37) are defined in a statement of the same form as

Rate of ξ increase of fluid element	+	Net rate of ξ flow out of element	=	Rate of ξ increase due to diffusion	+	Rate of ξ increase due to sources

The transport equation for (ξ) is simply a reapplied application of the already established form of conservation laws, which alleges the sum of the rate of change and convective terms on the left-hand side is equal to the sum of the diffusive and source terms on the right-hand side. The (Γ) term in the transport equation represents the diffusion coefficient and Equation (2.37) is nearly always used as the beginning point for computational evaluations using the finite-volume method (or FVM).

Integration of Equation (2.37) over all three dimensions of a control volume (CV) is a critical step of the FVM (Versteeg and Malalasekera, 1995). Accordingly, the integration of this equation yields

$$\int_{CV} \frac{\partial(\rho\xi)}{\partial t} dV + \int_{CV} div(\rho\xi \boldsymbol{u})\, dV = \int_{CV} div(\Gamma\ grad\ \xi)\, dV + \int_{CV} S_\xi\, dV \tag{2.38}$$

Using Gauss' divergence theorem, the volume integrals of the convective term (second term on left-hand side) and diffusive term (first term on right-hand side) are rewritten as surface integrals over the bounding surface of the control volume. Applying this theorem for a vector ($\boldsymbol{a}$), Versteeg and Malalasekera (1995) define a general expression as

$$\int_{CV} div(\boldsymbol{a})\, dV = \int_{A} \boldsymbol{n} \cdot \boldsymbol{a}\, dA \tag{2.39}$$

where, ($\boldsymbol{n}\cdot\boldsymbol{a}$) is the component of vector ($\boldsymbol{a}$) in the direction of vector ($\boldsymbol{n}$) normal to surface element (dA). Hence, the integration of the divergence of vector ($\boldsymbol{a}$) over a volume is equal to the component of vector ($\boldsymbol{a}$) normal to the boundary surface of the volume integrated over the entire surface (A) bounding the volume (Versteeg and Malalasekera, 1995). A formal definition and specific example of Gauss' theorem is provided in detail by Spiegel and Liu (1999), which is illustrated in Figure 2.5 below.

THE DIVERGENCE THEOREM

Let S be a closed surface bounding a region of volume V; then if $\mathbf{N}$ is the positive (outward drawn) normal and $d\mathbf{S} = \mathbf{N}\,dS$, we have [see Fig.]

$$\int_V \nabla \cdot \mathbf{A}\, dV = \int_S \mathbf{A} \cdot d\mathbf{S}$$

The result is also called *Gauss' theorem* or *Green's theorem*.

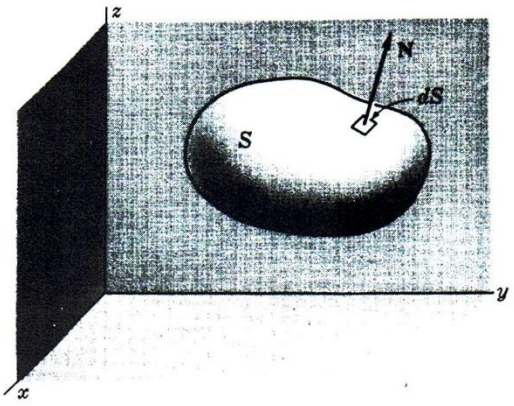


Figure 2.5: Definition and example illustration of Gauss' divergence theorem; adapted in part from (Spiegel and Liu, 1999).

Therefore, applying Gauss' divergence theorem in the form of Equation (2.39) to both the convective and diffusive term volume integrals, Equation (2.38) becomes

$$\int_{CV} \frac{\partial(\rho\xi)}{\partial t}\, dV + \int_A \boldsymbol{n} \cdot (\rho\xi \boldsymbol{u})\, dA = \int_A \boldsymbol{n} \cdot (\Gamma\, grad\, \xi)\, dA + \int_{CV} S_\xi\, dV \qquad (2.40)$$

This achieves the critical task of rewriting the convection and diffusion terms in Equation (2.38) as integrals over the bounding or outermost surface extents of the control volume rather than the entire volume itself; thereby, providing a method of imposing boundary and/or initial conditions of which the direction of gradients is known. Propagating information is subsequently followed throughout the interior solution domain (or flow field) as part of the iterative calculation procedure. A special case of particular importance involves problems characterized by a steady-state solution field. In these problems, the rate of change term in Equation (2.40) for the total amount of (ξ) in the control volume (first term on left-hand side) is equal to zero. This yields a simplified form of Equation (2.40), which is written as

$$\int_A \boldsymbol{n} \cdot (\rho\xi \boldsymbol{u})\, dA = \int_A \boldsymbol{n} \cdot (\Gamma\, grad\, \xi)\, dA + \int_{CV} S_\xi\, dV \qquad (2.41)$$

Hence, Equation (2.41) is the integrated form of the steady-state transport equation. In summary, the above derivation emphasizes that the partial differential equation of a fluid property (ξ), when properly integrated, results in a conservation statement of (ξ) for a finite control volume of interest.

General information that is specific to the internal numerical solver in FLUENT CFD software is included within the FLUENT user's guide beginning in Chapter 25

(FLUENT Inc., 2006). Analogous to the derivation of Equation (2.40) above, the governing conservation equations in FLUENT are collectively cast in the form of a general transport equation for a scalar quantity (ϕ). The specific governing equations for conservation of mass, momentum, and energy as well as any scalar property are obtained by selecting appropriate values for the variable (ϕ), diffusion coefficient, and source terms in the general transport equation. In order to solve the governing flow equations, FLUENT employs a 'control-volume-based' technique in which the domain is sectioned into discrete control volumes using a computational grid and the governing equations are integrated over the individual control volumes to obtain algebraic equations (*i.e.* a finite-volume approach) for the discrete dependent variables. The unsteady conservation equation for transport of a scalar quantity (ϕ), written in integral form for an arbitrary control volume (V), is given in the FLUENT user's guide as (FLUENT Inc., 2006):

$$\int_V \frac{\partial \rho \phi}{\partial t} dV + \oint \rho \phi \vec{v} \cdot d\vec{A} = \oint \Gamma_\phi \nabla \phi \cdot d\vec{A} + \int_V S_\phi \, dV \tag{2.42}$$

Despite slight differences in the symbolism and mathematical notations used in presenting Equations (2.40) and (2.42), it is clear that both forms of the derived general transport equation are equivalent. Obviously, this result is expected since Equations (2.40) and (2.42) are both formulated based on fundamental physical principles from the conservation laws of physics expressed on an integral control volume basis.

A final topic of consideration warrants brief discussion prior to the provision of any further details regarding the FLUENT solver or modeling options, as no solution procedure exists otherwise for the characteristic flows examined in this research. In other words, consider a flow in which the assumptions of incompressibility and constant

viscosity are valid. In this case, the governing flow equations including the continuity and Navier-Stokes (or momentum) equations are given by Equations (2.6) and (2.34) through (2.36), respectively. Note that assumptions of constant fluid density and viscosity completely decouple the energy equation from the analysis. As a result, the implication is that only the four governing flow equations are necessary to resolve the desired pressure and velocity solution fields since these equations are self-contained; there are four equations and four dependent unknown variables (u, v, w, and p).

However, the appearance of performing a few simple calculations in order to determine values for these dependent variables is superficial at best. In fact, a straightforward numerical technique for obtaining a solution of the incompressible flow equations does not exist; there are no 'natural' equations to solve for pressure (Ball, 2004). Therefore, a relationship for the coupling between pressure and velocity adjustments to enforce the mass conservation constraint as part of an iterative strategy for acquiring the correct pressure field is critical in achieving a solution. This difficulty is transcended by an iterative process termed the pressure correction technique, proposed originally by Patankar and Spalding (1972). The philosophy of the pressure correction method with the implementation of a pressure correction formula is embodied in a numerical algorithm commonly referred to as Patankar's semi-implicit method for pressure-linked equations (SIMPLE).

Further explanation of this algorithm is provided by introducing a simplified case of the above fluid flow conditions in a primitive variable formulation (*i.e.* non-conservation form of governing equations in which the dependent solution variables are exactly the unknown primitive variables sought). In this example, the incompressible and constant viscosity fluid assumptions are maintained, and the body force terms in the

momentum component equations are negated for simplicity. Therefore, the continuity equation under consideration is

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}+\frac{\partial w}{\partial z}=0 \tag{2.43}$$

and the Navier-Stokes (or momentum) equations, with velocity gradients for stresses compacted using the Laplacian operator, become

$$\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial x}+\upsilon\nabla^2 u \tag{2.44}$$

$$\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial y}+\upsilon\nabla^2 v \tag{2.45}$$

$$\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}=-\frac{1}{\rho}\frac{\partial p}{\partial z}+\upsilon\nabla^2 w \tag{2.46}$$

As previously mentioned, the appearance of the governing flow equations [Equations (2.43) through (2.46)] for an incompressible, constant viscosity fluid flow induces the impression of a seemingly straightforward approach for obtaining a solution to the set of four equations and four dependent unknowns. However, since these equations are intricately coupled and no transport or other equation of state exists for the pressure field, an iterative pressure correction method is required in order to solve the equations.

The implementation of this method as a solution procedure is realized in the form of Patankar's SIMPLE numerical algorithm. Referencing Patankar (1981), the SIMPLE procedure for obtaining a solution to the equations under consideration is described by Ball (2004), which is summarized as follows:

1. Arbitrarily choose values for the entire pressure field to start the iterative process; denote the guessed pressures by (p^*).
2. Substituting (p^*) values into Equations (2.44), (2.45), and (2.46), solve for the velocity components; denote these values by (u^*, v^*, and w^*).
3. Plug the resulting velocities (u^*, v^*, and w^*) into Equation (2.43), noting that these velocities which were based on a guessed set of (p^*) values for the pressure field in step 1 will not necessarily satisfy the continuity equation. Therefore, using Equation (2.43), derive a pressure correction (p') such that the sum of (p') and (p^*) brings the velocity field closer into agreement with the conservation of mass constraint. In other words, the corrected pressure (p) is defined as

$$p = p^* + p' \tag{2.47}$$

4. Using (p'), obtain corresponding velocity corrections (u', v', and w') such that

$$u = u^* + u' \tag{2.48}$$

$$v = v^* + v' \tag{2.49}$$

$$w = w^* + w' \tag{2.50}$$

5. Solve for other necessary physical quantities (*i.e.* temperature) if they influence the fluid flow through density, viscosity, or source terms.

6. Substitute the new value (p) from Equation (2.47) as the new value of (p^*). Return to step 2, and continue repeating the iterative process until the velocity field determined satisfies the continuity equation (at which point the solution is converged).

The SIMPLE procedure outlined above emphasizes the fact (p) in the Navier-Stokes equations is simply a Lagrange multiplier which is used to enforce an external constraint, namely incompressibility (Ball, 2004).

Taking the analysis a step further, the application of the SIMPLE procedure is briefly demonstrated by considering a generalized pseudo-calculation (or walkthrough) of the example case above, adapted in large part from the primitive variable formulation discussion in Ball (2004), but under the additional assumption of two-dimensional flow. Hence, initially reflecting on the x-direction only, the x-component momentum equation [Equation (2.44)] in two-dimensional form with a semi-implicit differencing method is given by

$$\frac{u^{n+1}-u^n}{\Delta t}+u^n(\delta_x^n u)+v^n\left(\delta_y^n u\right)=-\frac{1}{\rho}\frac{\partial p}{\partial x}\Big|^{n+1}+v\left(\delta_{xx}^n u+\delta_{yy}^n u\right) \tag{2.51}$$

where, (δ_x, δ_y, δ_{xx}, and δ_{yy}) are appropriate difference operators. For example, using central differences gives the operators in one of the following two subscript forms

$$\delta_x^n u=\frac{u_{i+1,j}^n-u_{i-1,j}^n}{2\Delta x} \tag{2.52}$$

$$\delta_{yy}^n u=\frac{u_{i,j+1}^n-2u_{i,j}^n+u_{i,j-1}^n}{(\Delta y)^2} \tag{2.53}$$

Remaining difference operators are similarly defined. The iterative process is started assuming the correct velocity and pressure values are in hand, and initially providing a guess for the pressure field (p^{n+1}).

Thus, substituting the guessed pressures into Equation (2.51), the expression for the x-component momentum equation based on these values becomes

$$\frac{u^{*n+1} - u^n}{\Delta t} + u^n(\delta_x^n u) + v^n\left(\delta_y^n u\right) = -\frac{1}{\rho}\frac{\partial p^*}{\partial x}\Big|^{n+1} + v\left(\delta_{xx}^n u + \delta_{yy}^n u\right) \tag{2.54}$$

Subsequently, if Equation (2.51) represents the correct velocity and pressure fields, subtracting Equation (2.54) from Equation (2.51) gives

$$\frac{u^{n+1} - u^{*n+1}}{\Delta t} = -\frac{1}{\rho}\left[\frac{\partial p}{\partial x}\Big|^{n+1} - \frac{\partial p^*}{\partial x}\Big|^{n+1}\right] \tag{2.55}$$

Substituting Equations (2.47) and (2.48) into Equation (2.55), and simplifying yields

$$u' = -\frac{\Delta t}{\rho}\frac{\partial p'}{\partial x} \tag{2.56}$$

Therefore, Equation (2.56) is the x-component velocity correction at time step (*n+1*). A similar analysis is followed for obtaining the y-component velocity correction at time step (*n+1*), which is given by

$$v' = -\frac{\Delta t}{\rho}\frac{\partial p'}{\partial y} \tag{2.57}$$

The pressure correction is derived from the continuity equation [Equation (2.43)] written in two-dimensional form as

$$\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0 \tag{2.58}$$

Substituting Equations (2.48) and (2.49) into Equation (2.58), the equation becomes

$$\frac{\partial(u^*+u')}{\partial x}+\frac{\partial(v^*+v')}{\partial y}=0 \tag{2.59}$$

and rearranging the terms in meaningful groups,

$$\left(\frac{\partial u^*}{\partial x}+\frac{\partial v^*}{\partial y}\right)+\left(\frac{\partial u'}{\partial x}+\frac{\partial v'}{\partial y}\right)=0 \tag{2.60}$$

Replacing the terms (u') and (v') above with an equivalent expression from Equation (2.56) and Equation (2.57), respectively, Equation (2.60) is rewritten as follows

$$\left(\frac{\partial u^*}{\partial x}+\frac{\partial v^*}{\partial y}\right)+\left[\frac{\partial}{\partial x}\left(-\frac{\Delta t}{\rho}\frac{\partial p'}{\partial x}\right)+\frac{\partial}{\partial y}\left(-\frac{\Delta t}{\rho}\frac{\partial p'}{\partial y}\right)\right]=0 \tag{2.61}$$

Thus, with slight rearrangement and use of compact vector notation, the final form of Equation (2.61) is given by

$$\nabla^2 p'=\frac{\rho}{\Delta t}(\nabla\cdot\boldsymbol{V}^*) \tag{2.62}$$

The pressure correction is obtained by solving for (p') in Equation (2.62), which expresses a balance for satisfying the constraint between the guessed velocity field and the conservation of mass.

It is important to realize that Equation (2.62) is nothing more than a Poisson equation for the pressure correction. The Poisson equation is an elliptic equation, consistent with the flow-field characteristics of an incompressible fluid flow in which pressure disturbances propagate throughout the entire flow domain (Anderson, 1995). Therefore, in order to solve Equation (2.62), a boundary condition associated with (p') specified over the complete boundary encapsulating the computational domain is required. The strategy employed in determining this pressure correction boundary condition is extremely dependent upon the specifics of the physical problem being evaluated. Discussions and examples related to the derivation of boundary conditions consistent with the philosophy of the pressure correction method, as well as other considerations such as the use of staggered grids or upwind schemes, are presented in the literature elsewhere (Anderson, 1995; Versteeg and Malalasekera, 1995). However, with a specified boundary condition associated the pressure correction, Equation (2.62) is solved for (p') which subsequently permits calculation of the velocity correction from Equations (2.56) and (2.57). As a result, the correct pressure and velocity fields (*i.e.* the new guessed values) are obtained from Equation (2.47) along with Equations (2.48) and (2.49), respectively, and the SIMPLE procedure outlined above is repeated until ($\boldsymbol{V}^*$) satisfies the continuity equation (*i.e.* the pressure correction is zero).

Although there are obvious dissimilarities in the discretized equations utilized in a finite-difference primitive variable formulation (as outlined in the simplified example above) and an equivalent finite-volume approach in conservation equation form as

utilized in the FLUENT CFD software, the thought process associated with implementing the SIMPLE solution algorithm in the two approaches are akin. The simplified primitive variable formulation is chosen for illustrating the application of this solution procedure because it is easy to follow and more straightforward than the equivalent approach employed in FLUENT. However, the essential application is similar enough that the paradigm of the SIMPLE procedure presented above is sufficient in providing an outline of the basic solution framework incorporated within the FLUENT numerical solver.

2.1.3 Turbulence

In analyzing the flow of fluidic elements through a specified domain construct, the flow is characterized as turbulent when all the transport quantities (*i.e.* mass, momentum, and energy) exhibit aperiodic, irregular fluctuations in both time and space, thereby enhancing the mixing of these transport variables as a result of such conditions (Chilka and Kulkarni, n.d.). The subject of modeling turbulent flows in CFD applications is one of enormous extent and the focus of widespread efforts intending to advance current modeling capabilities to more accurately account for the effects of turbulent flow phenomena. Although the developing flows examined in this research (*i.e.* expected Reynolds numbers spanning 70 to 900) are well-within accepted limits of the laminar flow regime, a universal, clear-cut criterion establishing the demarcations between laminar, transitional, and turbulent flows in all applicable cases does not exist. In fact, experimental observations of several internal, wall-bounded (or ducted) airflows reveal departing deviations from laminar flow behavior at Reynolds numbers in the range of ($Re \approx 1000$); with the formation of distinct vortices and flow separation in areas of minimal sudden expansion (Armaly *et al.*, 1983). Therefore, as several turbulence models are

employed in a few preliminary case trial evaluations in an attempt to quantify potential pressure-loss contributions due to transitional/turbulent flow effects, a brief discussion of the turbulence modeling capabilities in the FLUENT CFD software is warranted.

In general, there are three primary computational approach options to consider for modeling turbulence: (1) direct numerical simulation (DNS); (2) large eddy simulation (LES); and (3) Reynolds-averaged Navier-Stokes (RANS) equations models. Theoretically, the simulation of all turbulent flows is achievable by numerically solving the full time-dependent Navier-Stokes equations on a fine enough grid and with a sufficiently small time step that the entire spectrum of scales (in particle motion) is resolved and no modeling is required at all. It is simply a problem of solving the equations mathematically using an appropriate numerical technique. This approach is more commonly referred to as DNS and requires a truly phenomenal amount of computational resources to utilize, even for simple two-dimensional pipe flows (DNS is not available in FLUENT).

LES provides an alternative relief in computational resource requirements in comparison to the DNS approach by solving the 'spatially-averaged' (or filtered) time-dependent Navier-Stokes equations in which large eddies are directly resolved, but the smaller scale eddies are modeled (Bell, 2004). The filtering process effectively removes eddies of characteristic scales smaller than the filter width, which is typically the size of the grid spacing used in constructing the computational domain. Unfortunately, despite relaxed computing requirements by modeling the small-scale fluctuations, the amount of computational resources and efforts inherent in the application of LES remain extensive. The FLUENT CFD software package does include the capability to employ LES modeling under certain considerations, but impractical processing requirements

ultimately preclude LES as a realistic turbulence model solution for the majority of industrial functions.

As a result, the RANS-based approach represents the brunt force of turbulence models accessible within the FLUENT CFD software under most circumstances and constitutes plausible modeling solutions with regard to the current research analysis requirements. The RANS-based modeling approach fundamentally solves ensemble-averaged (or time-averaged) Navier-Stokes equations in which all turbulence scales are modeled; this substantially decreases the computational resource dedications necessary in order to model turbulent flows. FLUENT CFD software contains a number of different specifically formulated models within this genre of computational approach, characterized in a hierarchy of increasingly complex closure models. However, the amplified degree in which variables are modeled and not resolved introduces augmented uncertainty into the results as well as the added potential for numerical inaccuracies; therefore, careful consideration of the simulated flow field and a thorough understanding of the capabilities and limitations of the various model options is absolutely critical. An illustration of the relationship between the three primary computational approaches for modeling turbulence is provided in Figure 2.6 below for reference.

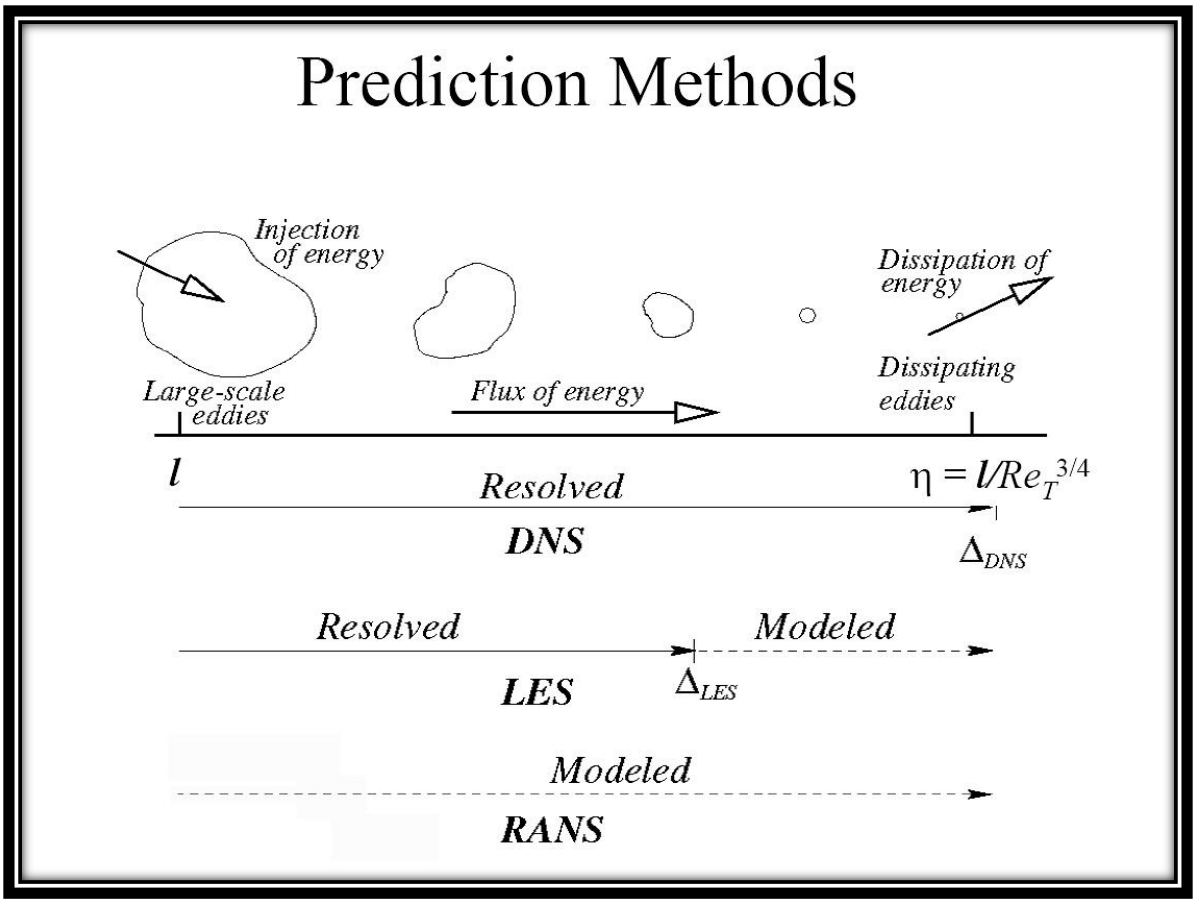


Figure 2.6: Comparison of the three primary computational approach options for modeling turbulence with associated range of eddy sizes (scales) resolved or modeled in each approach (FLUENT Inc., 2005).

In Reynolds averaging, the solution variables are decomposed in the instantaneous (or exact) Navier-Stokes equations into ensemble-averaged (or time-averaged) and fluctuating parts in which the velocity components are expressed as follows (FLUENT Inc., 2006; Ball, 2004):

$$u_i = \overline{u}_i + u_i' \tag{2.63}$$

where, ($\overline{u_i}$) and (u_i') are the mean and fluctuating parts of the velocity components (i = 1, 2, 3). Similarly, pressure and other scalar quantities are given by

$$\phi = \bar{\phi} + \phi' \tag{2.64}$$

where, (ϕ) represents any scalar such as energy, pressure, or concentration with equivalent mean and fluctuating parts as defined in Equation (2.63). Substituting expressions of the above form for flow variables in the instantaneous continuity and momentum equations, averaging over time, and dropping the over-bar on the mean velocity terms, yields the ensemble-averaged conservation equations as (FLUENT Inc., 2006)

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) = 0 \tag{2.65}$$

$$\frac{\partial}{\partial t}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j) = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial u_l}{\partial x_l}\right)\right] + \frac{\partial}{\partial x_j}\left(-\rho\overline{u_i' u_j'}\right) \tag{2.66}$$

Equations (2.65) and (2.66) are the full Reynolds-averaged Navier-Stokes (RANS) equations. These equations have the same general form as the instantaneous Navier-Stokes equations, but the velocities and solution variables are now represented as time-averaged values.

The key feature of the Reynolds-averaged approach for modeling turbulence is the method in which the Reynolds stresses ($-\rho\overline{u_i' u_j'}$) in Equation (2.66) are modeled. There are two primary options to consider: (1) the Boussinesq approach and (2) the

Reynolds stress transport equations embodied in the Reynolds stress model (RSM). In the RSM-based approach, transport equations for each of the terms in the Reynolds stress tensor are solved including an extra 'scale-determining' equation, which results in seven additional transport equations for three-dimensional flows. In most situations, models based on the alternative Boussinesq approach are sufficient, and the significant increase in computational expense associated with the RSM is not justified (FLUENT Inc., 2006). These 'eddy-viscosity' models employing the Boussinesq hypothesis constitute the bulk of turbulence models available within the FLUENT CFD software and offer the advantage of a relatively low computational cost requirement (Ball, 2004). Therefore, since the application of turbulence models is limited throughout this research and all implemented models are based solely on the Boussinesq hypothesis, the RSM is not discussed in further detail for reasons of brevity. However, additional information on the Reynolds stress model approach is available in the FLUENT user's guide if desired (FLUENT Inc., 2006).

In the Boussinesq approach, the Reynolds stresses in Equation (2.66) are related to the mean velocity gradients by a turbulent (eddy) viscosity (μ_t) as follows (FLUENT Inc., 2006):

$$-\rho\overline{u_i'u_j'} = \mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{2}{3}\left(\rho k+\mu_t\frac{\partial u_k}{\partial x_k}\right)\delta_{ij} \tag{2.67}$$

In other words, the Reynolds stresses are assumed proportional to the mean velocity gradients by a constant of proportionality (μ_t), an isotropic scalar quantity termed the turbulent (eddy) viscosity, the relation of which is drawn from analogy with molecular transport of momentum (Bell, 2004). The disadvantage of the approach (as presented)

resides in the assumption of (μ_t) as an isotropic scalar quantity, which is valid at the molecular level, but not strictly true at the macroscopic level. However, unless the anisotropy of turbulence has a dominant effect on the mean flow (i.e. situations include highly swirling flows and stress-induced secondary flows), the isotropic assumption is generally acceptable (FLUENT Inc., 2005).

The turbulence models under the Boussinesq approach are often classified based on the number of transport equations solved in modeling the turbulent viscosity (*e.g.* zero-, one-, two-…equation models). This naming convention more importantly identifies the number of independent scales used to represent or define (μ_t), which obviously is the term sought in correlating the Reynolds stresses to the mean flow in Equation (2.67) above. Typically, based on dimensional analysis, the turbulent viscosity (μ_t) is determined from a turbulence time scale (or velocity scale) and a length scale. The common scales used in popular turbulence models employing the Boussinesq approach are listed below (Bell, 2004).

- Turbulent kinetic energy (k) [L^2/T^2]:

$$k \equiv \frac{\overline{u_i' u_i'}}{2} \tag{2.68}$$

- Turbulence dissipation rate (ε) [L^2/T^3]:

$$\varepsilon \equiv \upsilon \overline{\frac{\partial u_i'}{\partial x_j}\left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i}\right)} \tag{2.69}$$

- Specific dissipation rate (ω) [1/T]:

$$\omega \sim \frac{\varepsilon}{k} \tag{2.70}$$

As a result, the turbulent viscosity (μ_t) is uniquely determined for each turbulence model depending upon the number of and specific independent scale(s) used in solving for the turbulent viscosity values. The calculation of (μ_t) in several of the more prominent Boussinesq-based turbulence models available in the FLUENT CFD software are summarized in the following list (Bell, 2004).

- Spalart-Allmaras model - a one-equation turbulence model that solves only one transport equation for a modified turbulent viscosity:

$$\mu_t \equiv f(\tilde{v}) \tag{2.71}$$

- Standard, Renormalization-group, and Realizable (k-ε) models - are two-equation turbulence models that solve transport equations for (k) and (ε):

$$\mu_t \equiv f\left(\frac{\rho k^2}{\varepsilon}\right) \tag{2.72}$$

- Standard and Shear-stress transport (k-ω) models - are two-equation turbulence models that solve transport equations for (k) and (ω):

$$\mu_t \equiv f\left(\frac{\rho k}{\omega}\right) \tag{2.73}$$

Although the discussion above is only a very simple introduction to the turbulence modeling options available within the FLUENT CFD software, it serves the intended purpose of outlining the basic methods and considerations specific to the few turbulence models employed in this research. A general knowledge and understanding of these turbulence models is pertinent for the comprehension of certain CFD trial simulations and results discussed later in Chapter 5 of this dissertation. In summary, the key features and appropriate applications associated with commonly used RANS-based turbulence models available in the FLUENT software are illustrated in Figure 2.7 below.

Spalart-Allmaras Model	One-equation model	Designed specially for aerospace applications, involving wall-bounded high speed flows.
Standard $k-\varepsilon$ Model	Simplest of two-equation models	Robust. Suitable for initial iterations.
RNG $k-\varepsilon$ Model	– Variant of standard $k-\varepsilon$ – Has an additional term in ε equation.	Accurate for rapidly strained and swirling flows.
Realizable $k-\varepsilon$ Model	– Variant of Standard $k-\varepsilon$ model – New formulation for turbulent viscosity – New transport equation for ε	Accurate for spreading of both planar and rounded jets. Recommended for flows with boundary layers under strong adverse ∇p, separation and recirculation.
Standard $k-\omega$ Model	Solves for $k-\omega$ ω = Specific dissipation rate (ε/k)	Recommended for low-Re flows, wall bounded boundary layer, and for transitional flows.
SST $k-\omega$ Model	– Variant of Standard $k-\omega$ model – Behaves like $k-\omega$ in near wall region – Behaves like standard $k-\varepsilon$ in the free stream	More accurate and reliable for a wider class of flows, like adverse ∇p in airfoils, transonic shock waves, etc.
Reynolds Stress Model	– Five-equation model – Avoids isotropic formulation of turbulent viscosity	Suitable for complex 3D flows with strong swirl/rotation. Run time and memory intensive.

Figure 2.7: Key features and applications of commonly used RANS-based turbulence models available in the FLUENT CFD software (Chilka and Kulkarni, n.d.).

On a final note, accurate near-wall modeling is extremely important in most engineering applications since walls serve as the main source of vorticity and turbulence within a given flow field. In particular, the successful prediction of pressure drop for internal fluid flows as focused upon in this research, depends on the fidelity of local wall shear predictions. Unfortunately, distinct near-wall modeling techniques are necessary since near-wall flows are anisotropic due to the presence of walls and equations cannot be

integrated all the way down to the wall surface. Therefore, a brief review of the near-wall modeling options available and their appropriate implementation with specific turbulence models is a requisite in the discussion of turbulent flows.

Numerous documented flow experiments with the assistance of dimensional analysis confirm the existence of a multilayer inner region and single-layer outer region, which compose the turbulent boundary layer adjacent to a solid surface. Within the inner region (up to 20% of the total wall layer thickness), the shear stress in the fluid is nearly constant and equal in magnitude to the wall shear stress. The three layers that constitute the inner region are characterized (from nearest to the wall surface) as follows: (1) viscous linear sublayer, where viscous stresses dominate the adjacent wall flow; (2) buffer layer, in which turbulent and viscous stresses have similar magnitude; and (3) log-law layer, where turbulent Reynolds stresses are dominant (Versteeg and Malalasekera, 1995). The outer region is dependent upon the mean flow, which is free from direct viscous effects and characterized by inertial forces. These structural sublayers composing the inner and outer regions of the turbulent boundary layer are graphically represented in Figure 2.8 below.

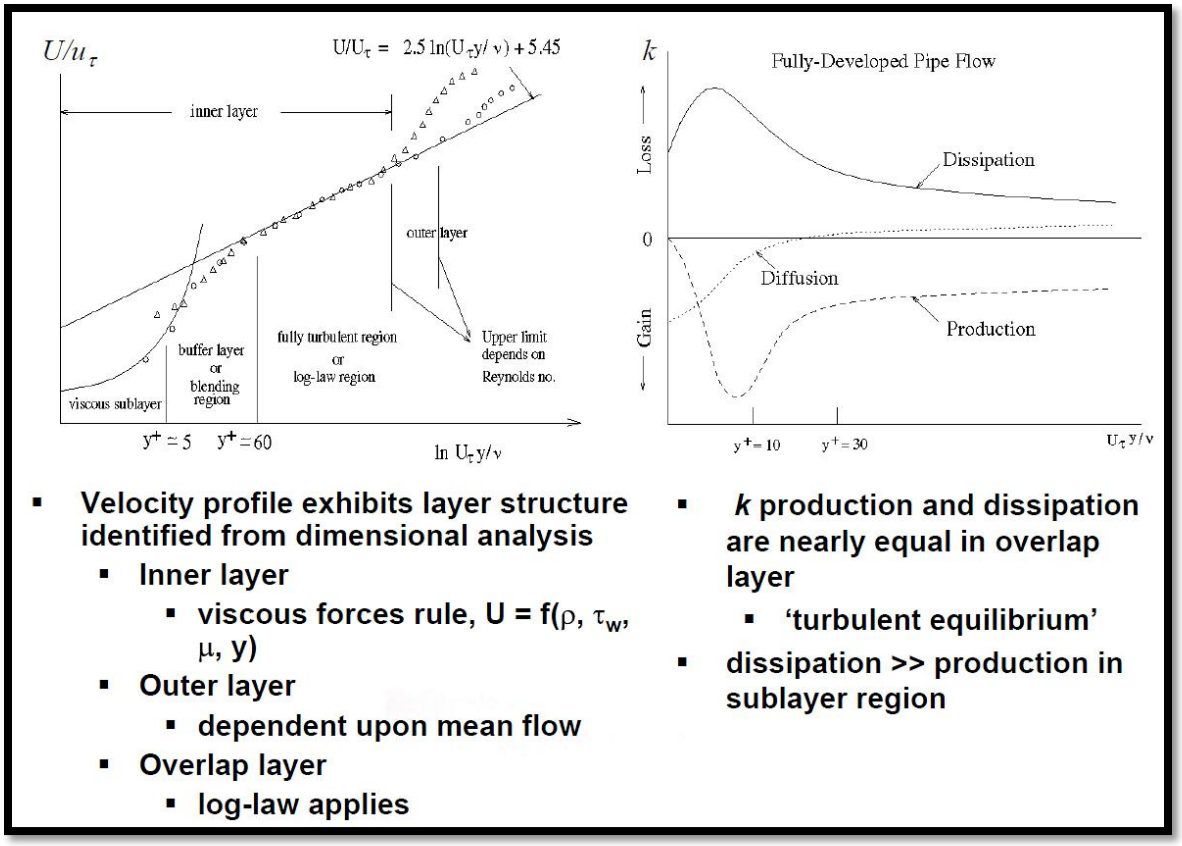


Figure 2.8: Flow behavior in near-wall turbulent boundary layer regions and characteristics associated with each sublayer (Bell, 2004).

The ultimate goal of near-wall modeling is to numerically reflect the structure of the turbulent boundary layer dictated in Figure 2.8 and reproduce the flow behavior as accurately as possible. Demarcations between the various sublayers are indicated clearly in the plots at specific values for the (y^+) term, which is a non-dimensional parameter referred to as the wall unit. The wall unit functions as a distance measurement between a cell centroid and the wall surface for wall-adjacent cells. Accordingly, the wall unit is formally defined as (FLUENT Inc., 2006)

$$y^+ \equiv \frac{\rho u_\tau y}{\mu} \tag{2.74}$$

where, (u_τ) is the friction velocity given by

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{2.75}$$

Therefore, the wall unit defined by Equation (2.74) provides upper and lower reference bounds for distance between a wall-adjacent cell centroid and the wall surface for each sublayer in the turbulent boundary layer.

The correct modeling of certain flow attributes associated with each region of the turbulent boundary layer, which are deemed important in the final solution, requires the appropriate selection of a turbulence model, a suitably paired near-wall modeling approach, and a sufficient grid resolution pertaining to the near-wall treatment option employed. In general, the near-wall modeling options are simply comprised of 'wall functions', which serve as a collection or 'set of laws' providing boundary conditions for momentum, energy, mass species, and turbulence quantities (FLUENT Inc., 2005). There are two formulations available within the FLUENT CFD software for near-wall modeling; these include the basic wall function options and the enhanced wall treatment (or low-Re) option. The standard and non-equilibrium wall functions are part of the basic wall function options that are empirically based and designed for high-Re flows in which the viscosity affected, near-wall region is not resolved. Grid resolution requirements in the near-wall vicinity are relatively course for these options.

On the other hand, the enhanced wall treatment model (often termed the near-wall model approach since the alternative wall function options do not actually resolve the near-wall region) combines both enhanced wall functions and a two-layer zonal concept

for low-Re flows that is capable of resolving the viscous stress dominated near-wall region all the way down to the wall. However, this near-wall treatment option comes at the cost of a very fine near-wall mesh requirement and increased computational expense. An outline of both near-wall modeling approaches along with respective visual mesh representations are provided in Figure 2.9 below.

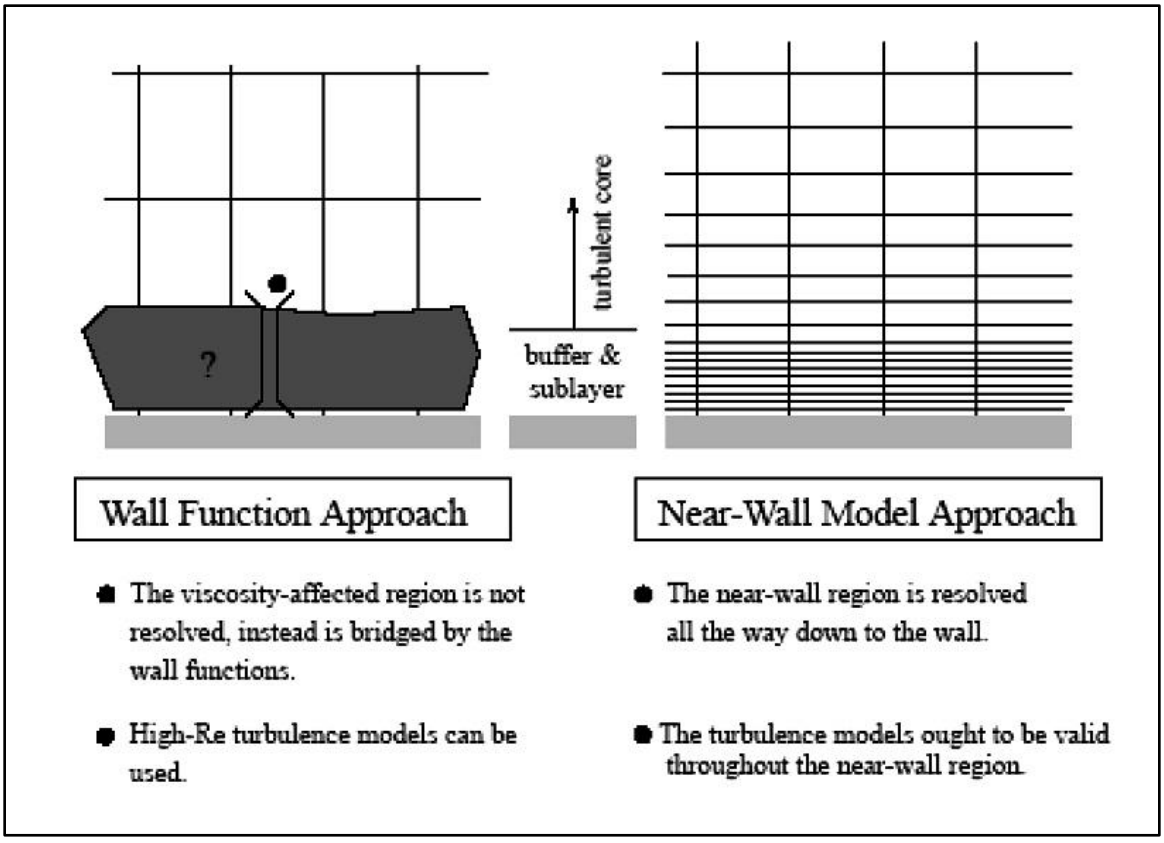


Figure 2.9: Comparison of mesh requirements and levels of resolved viscosity-affected region in each of the two near-wall modeling options (FLUENT Inc., 2006).

In conclusion, based on the key flow features expected to develop within the computational domain evaluated, the initial task is to identify the most appropriate turbulence model available within the FLUENT CFD software package. Subsequently, depending upon the turbulent boundary region/sublayer nearest to the wall that needs

resolved, the location (wall unit distance) of the first wall-adjacent cell is determined (noting this is initially an approximation since the actual value is solution dependent). In other words, to capture flow characteristics in the viscous sublayer (*i.e.* for accurate prediction of frictional drag, internal pressure drop, flow separation, *etc.*), the enhanced wall treatment option is necessary which requires a wall unit value within a specified range for model accuracy (max $y^+ \lesssim 5$). As an alternative, standard wall functions are applicable if the desired flow resolution begins outside the buffer layer and within the log-law layer, as in pervasive, high-Re turbulent core flows. Again, since the (y^+) values are solution dependent, the grid resolution in the near-wall vicinity occasionally requires multiple adaptations in order to place the first wall-adjacent cell layer at the correct wall unit distance required by the near-wall modeling option utilized. An overview of the above considerations is presented in Figure 2.10 below, including acceptable ranges of wall unit values for each approach.

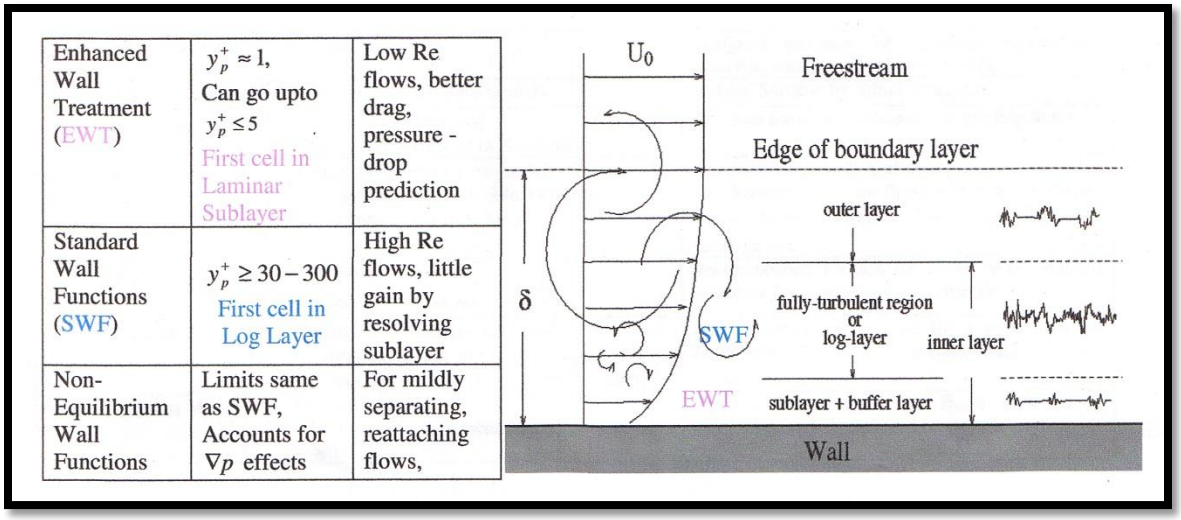


Figure 2.10: Mesh requirements and typical areas of application for near-wall modeling options available in FLUENT CFD software (Chilka and Kulkarni, n.d.).

Important aspects to remember with regard to the mesh generation and near-wall treatment stem from the dependency of obtaining an accurate solution based on both the turbulence model selection and the required wall unit distance for proper near-wall boundary modeling, per acceptable range of (y^+) values. In addition, pairing of the turbulence model and near-wall treatment option is dictated by the flow attributes within the near-wall region desired for capture.

2.1.4 Buoyancy

The adding or removing of heat energy in most fluids causes the associated density of the fluid to fluctuate with the changing temperature, which often induces a flow due to the force of gravity acting on these density variations within the fluid domain. Although the inclusion of buoyancy forces in a simulated fluid flow is relatively straightforward, especially within the FLUENT CFD software, there are a few important considerations of noteworthy significance that warrant concise deliberation. However, the scope of the following discussion is limited to the particular formulation for buoyancy forces that is utilized in all heated flow simulations performed in this research. As a result, there are two subjects of principal focus: (1) treatment of buoyancy forces in the momentum equations and (2) calculation of fluid density variations within the flow field.

The body force in each directional component momentum equation appears as part of a conglomerate source term embodying the buoyancy force in that direction; expansion of this term reveals a specific buoyancy (or body-force) term that represents this force. The form of the buoyancy term is dependent upon whether a Boussinesq model or temperature function for fluid density is employed. Due to the large temperature gradients expected to develop within the computational domain, the

Boussinesq model is rendered invalid and the latter temperature function approach is implemented in all model setups. Therefore, the form of the body-force term in the momentum equations is written as (FLUENT Inc., 2006)

$$f_{body-term} \equiv (\rho - \rho_0)g \tag{2.76}$$

where, (ρ_o) is the operating density. This form stems from the redefinition of pressure in FLUENT, which is given by

$$p_s' = p_s - \rho_0 g x \tag{2.77}$$

where, (p_s') is the modified static pressure that includes the hydrostatic head, (p_s) is the actual static pressure, and (x) is the position component in the gravity (g) applied direction. Hence, pressure boundary inputs do not include hydrostatic pressure differences and the inherent hydrostatic pressure in a fluid at rest is

$$p_s' = 0 \tag{2.78}$$

As discussed further in Chapter 5, the surrounding fuel assembly environment in heated flow simulations, which is not explicitly modeled as part of the computational domain, is assumed infinite with constant ambient air conditions maintained. Therefore, buoyancy forces driving the internal assembly airflow, which arise from the density differential at inner-assembly locations and the ambient environment, are manifest through the specification of a constant operating density value in Equation (2.76) equal to the ambient air density. Again, this simply avoids the need to include a large segment of

the surrounding fuel assembly volume in the computational domain. Furthermore, equivalent modified pressure values defined by Equation (2.77) are stipulated at pressure inlet and outlet boundaries of the fuel assembly as no externally imposed pressure gradients are present outside the modeled domain.

The calculation of fluid density throughout the buoyancy-induced flow field is a final topic of direct interest. Utilizing the specified operating density option and form of the body-force term expressed by Equation (2.76) in the momentum equations, implies a necessity to define the fluid density as a function of temperature. This is achieved using the incompressible ideal gas law, which is applicable in the case trials evaluated in this research due to the expectation of sufficiently minute pressure variations developing within the flow field; thereby, permitting the reasonable justification of fully incompressible flow. As a result, the ideal gas law is implemented in order to express the relationship between fluid density and temperature. However, in this form, the pressure term in the ideal gas law is a constant set equal to the operating pressure such that (aside from fluid temperature) density depends only on the constant operating pressure value and not the local relative pressure field. The formal expression for density based on the incompressible ideal gas law is written as (FLUENT Inc., 2006)

$$\rho = \frac{p_{op} M_w}{R_{gas} T} \tag{2.79}$$

where, (p_{op}) is the operating pressure. In the applicable heated flow simulations utilizing the incompressible ideal gas law, the operating pressure in Equation (2.79) is set equal to the ambient atmospheric pressure outside the fuel assembly.

2.2 BASIC SOLVER IMPLEMENTATION

A complete outline of the general CFD numerical framework on an introductory level necessitates further development of nominal extent to correlate relevant theory presented in Section 2.1 with extant numerical approaches and expressly formulated options within the FLUENT flow solver. The broadening of considerations with regard to this subject commences via reflection upon the general scalar transport equation derived in Section 2.1.2, which is equivalently expressed in a variant form by Equation (2.42) within the FLUENT user's guide. For reference, Equation (2.42) is conveniently stated again as follows:

$$\int_V \frac{\partial \rho \phi}{\partial t} dV + \oint \rho \phi \vec{v} \cdot d\vec{A} = \oint \Gamma_\phi \nabla \phi \cdot d\vec{A} + \int_V S_\phi \, dV$$

In FLUENT, the equation above [Equation (2.42)] is applied to each finite control volume (or cell) comprising the computational domain. The general discretization of this equation for a given cell yields (FLUENT Inc., 2006)

$$\frac{\partial \rho \phi}{\partial t} V + \sum_f^{N_{faces}} \rho_f \vec{v}_f \phi_f \cdot \vec{A}_f = \sum_f^{N_{faces}} \Gamma_\phi \nabla \phi_f \cdot \vec{A}_f + S_\phi V \tag{2.80}$$

where, (N_{faces}) is the number of faces enclosing the cell, (ϕ_f) is the quantity of (ϕ) convected through the (f) face, ($\vec{A}_f$) is the area of the (f) face, and (V) is the cell volume.

The scalar equations evaluated by the FLUENT flow solver are similar in form to Equation (2.80), which is applicable to three-dimensional, unstructured meshes composed of any constituent element type supported. The user exerts control over the specific solver formulation employed in resolving these scalar equations by selecting the

appropriate scheme/method for spatial discretization, evaluation of gradients and derivatives, and temporal discretization required to calculate the scalar quantities at cell centers (ϕ) and/or cell faces (ϕ_f) in Equation (2.80). Multiple scheme/method options are available for deriving each of the unspecified scalar terms in Equation (2.80) and the reasonable selection of these options is notably dependent upon the problem assessed. Therefore, an overview pertaining to the requisiteness of these schemes/methods within the FLUENT CFD software as well as the predominantly implemented options in the research trials conducted is précised in the bulleted list below.

- **Spatial Discretization:** Discrete values of the scalar quantity (ϕ) are stored at the center of volume cells by default within the FLUENT CFD software. However, the convection terms in Equation (2.80) dictate values of the scalar at cell faces (ϕ_f). Hence, the face values are interpolated from the quantities at volume cell centers using an upwind scheme; (ϕ_f) is determined from values in the volume cell upstream relative to the normal velocity direction. As an aside, it is noteworthy to mention that diffusion terms in Equation (2.80) are central-differenced in FLUENT by default and always second-order accurate.

  In order to maintain second-order accuracy for remaining convection terms, the second-order upwind scheme is utilized for calculating the cell face values (ϕ_f), which is based on a multidimensional linear reconstruction method (FLUENT Inc., 2006). This achieves a higher-order of accuracy for cell face values via a Taylor series expansion of the solution at the cell center, around the centroid of the cell. Denoting (ϕ), ($\nabla\phi$), and ($\vec{r}$) as the cell-centered value, gradient in the upstream cell, and displacement vector between the face and upstream cell

centroid, respectively; the expression for the face value (ϕ_f) becomes (FLUENT Inc., 2006)

$$\phi_{f,SOU} = \phi + \nabla\phi \cdot \vec{r} \tag{2.81}$$

- **Gradients and Derivatives:** The evaluation of the gradient term ($\nabla\phi$) in each cell is necessary in order to exploit the second-order upwind scheme formulation represented by Equation (2.81). In addition to reconstructing scalar values at cell faces, gradients are essential for computing velocity derivatives and secondary diffusion terms as well. The method used for the calculation of gradients is based on the divergence theorem (or Green-Gauss theorem) which, for the gradient of a scalar (ϕ) at the cell center (*c0*), is given in discrete form by

$$(\nabla\phi)_{c0} = \frac{1}{V}\sum_{f}^{N_{faces}} \tilde{\phi}_f \vec{A}_f \tag{2.82}$$

where, ($\tilde{\phi}_f$) is the value of (ϕ) at the cell face centroid. Specifically, the Green-Gauss node-based gradient evaluation is used for all simulations performed in this research in which ($\tilde{\phi}_f$) is determined in Equation (2.82) by taking the average of the nodal values on the face as

$$\tilde{\phi}_f = \frac{1}{N_f}\sum_{n}^{N_f} \tilde{\phi}_n \tag{2.83}$$

where, (N_f) is the number of nodes on the face. The nodal values ($\tilde{\phi}_n$) in Equation (2.83) are evaluated from a weighted average of the cell values surrounding the nodes, reconstructing exact values of a linear function at a node by solving a constrained minimization problem (FLUENT Inc., 2006). As a result, this scheme preserves second-order spatial accuracy even on unstructured meshes.

- **Temporal Discretization:** Transient simulations require the integration of the transient terms (first group of terms on the left-hand side) in Equation (2.80) over a time step (Δt). Temporal discretization also includes the integration of each term in the governing differential flow equations over the same time step as well. However, all case trials in this research involve steady-state flow calculations and the governing equations are not discretized in time; consequently, the transient group of terms in Equation (2.80) are zero.

Using the spatial and temporal discretization procedures outlined above, as well as the gradients and derivatives evaluation method prescribed, the unknown terms requiring consideration in Equation (2.80) are sufficiently resolved in order to attain a solution to the generic scalar transport equation. In addition, Equation (2.80) is often observed as nonlinear in relation to the unknown scalar quantity (ϕ) at the cell volume center as well as in adjacent surrounding cells. Thus, a linearized form of Equation (2.80), as defined in the FLUENT user's guide, is given instead by

$$a_P \phi = \sum_{nb} a_{nb} \phi_{nb} + b \tag{2.84}$$

where, the subscript (nb) represents neighboring cells, (a_P) is the linearized coefficient for (ϕ), and (a_{nb}) is the linearized coefficient for (ϕ_{nb}).

Equations similar to Equation (2.84) are prescribed for each cell within the computational grid, which represents a system of algebraic expressions with a sparse coefficient matrix. The FLUENT flow solver calculates this linear system for scalar equations by employing a point implicit, Gauss-Seidel linear equation solver in juxtaposition with an algebraic multi-grid (AMG) solution method (FLUENT Inc., 2006). Specific details regarding the AMG solution method are not discussed in this dissertation, however, further information is provided in Chapter 25 of the FLUENT user's guide if desired (FLUENT Inc., 2006).

The general computational procedure discussed above for the transport of scalar quantities within the FLUENT flow solver is applicable to both the pressure-based and density-based approach options available for use as the distinct numerical solver technique employed in the fluid flow calculations. However, all simulation trials conducted in this research utilize the pressure-based implementation of the flow solver. Therefore, in concluding the discussion on basic solver performance, several practices directly related to the discretization and solution of the momentum and continuity equations by course of the pressure-based flow solver are briefly addressed. These are most easily illustrated in the descriptions that follow by contemplating the steady-state continuity and momentum governing flow equations as respectively written in the following integral form (FLUENT Inc., 2006):

$$\oint \rho \vec{v} \cdot d\vec{A} = 0 \tag{2.85}$$

$$\oint \rho\vec{v}\vec{v} \cdot d\vec{A} = -\oint p\boldsymbol{I} \cdot d\vec{A} + \oint \bar{\bar{\tau}} \cdot d\vec{A} + \int_V \vec{F}\, dV \tag{2.86}$$

where, ($\boldsymbol{I}$) is the identity matrix, ($\bar{\bar{\tau}}$) is the stress tensor, and ($\vec{F}$) is the force vector.

Focusing initially on the discretization of the momentum equations [Equation (2.86)], the scheme used for a general scalar transport equation (as mentioned above) is also exploited in the discretization of (for example) the x-component of Equation (2.86) by setting (ϕ) equal to (u), obtaining the expression

$$a_P u = \sum_{nb} a_{nb} u_{nb} + \sum p_f A_f \cdot \hat{\imath} + S \tag{2.87}$$

The discrete form of the continuity equation [Equation (2.85)] is obtained in a rather straightforward manner by simply integrating Equation (2.85) over a given control volume, which yields

$$\sum_{f}^{N_{faces}} J_f A_f = 0 \tag{2.88}$$

where, (J_f) is the mass flux ($\rho\vec{v}_n$) through the (f) face.

The primary points of interest with regard to the pressure-based solver approach in evaluating Equations (2.87) and (2.88) stem from the understanding that the pressure field and face mass fluxes in these equations are not known *a priori*, and therefore must be obtained as part of the solution process. In addition, as with the storage of scalar quantities (ϕ), FLUENT uses a co-located scheme in which pressure and velocity values are both stored at cell volume centers by default. Again, the pressure and velocity values

at adjoining faces between cells are required in Equations (2.87) and (2.88), respectively; hence, an interpolation scheme is necessary in order to calculate these face values. A brief summary of the pressure-based solver formulation, as it pertains to the relevant numerical techniques and optional schemes most-widely employed in this research for addressing the aforementioned issues, is outlined accordingly in the bulleted list below.

- **Pressure Interpolation:** The standard pressure interpolation scheme is the default scheme in FLUENT used to interpolate the pressure face values (p_f) from the cell centers in Equation (2.87). This scheme is based on a procedure proposed by Rhie and Chow (1983) in which the pressure value at the face between two cells (*c0*) and (*c1*) is determined by associated momentum equation coefficients as follows

$$p_f = \frac{\frac{p_{c0}}{a_{P,c0}} + \frac{p_{c1}}{a_{P,c1}}}{\frac{1}{a_{P,c0}} + \frac{1}{a_{P,c1}}} \tag{2.89}$$

  where, (a_P) is the momentum equation coefficient and (p) is the pressure at the cell center. Although the standard scheme is applied in a limited number of preliminary case trials, this procedure is problematic in the presence of body forces or curvature. Therefore, the second-order pressure interpolation scheme is the primary option exercised in the case studies performed including all final unheated simulation trials and reported results (the body-force-weighted scheme is used in heated trials and is discussed further in Chapter 5). The second-order scheme reconstructs the pressure at cell faces in a method analogous to the

second-order upwind of convection terms in the scalar equations (see above). This ensures second-order accuracy for pressure values at cell faces.

- **Velocity Interpolation:** Unlike the treatment of pressures at cell faces, there are no user selectable options available in the FLUENT software for the interpolation of required face velocities from cell-centered values in Equation (2.88). Again, the inherent procedure employed in FLUENT for the interpolation of the velocity at a cell face is similar to the method suggested by Rhie and Chow (1983), which performs momentum-weighted averaging with weighting coefficients based on (a_P) from Equation (2.87). Hence, using this approach, the face flux (J_f) is given by

$$J_f = \hat{J}_f + d_f(p_{c0} - p_{c1}) \tag{2.90}$$

where, (p_{c0}) and (p_{c1}) are the pressures at adjoining cell centers and ($\hat{J}_f$) is the weighted influence of velocities in these cells. The (d_f) term is a function of average momentum equation coefficient values (a_P) for the cells adjacent to the (f) face (FLUENT Inc., 2006).

- **Pressure-Velocity Coupling:** The pressure-velocity coupling is applied in order to introduce pressure into the continuity equation, which does not appear explicitly for incompressible flows in Equation (2.88) as the fluid density is not a direct function of pressure. This is accomplished by deriving an additional condition for the pressure field from Equation (2.90) as a result of recasting the continuity equation in a modified form; the pressure-velocity coupling option

elected by the user determines the specifics of the solution approach. SIMPLE is selected as the exclusive pressure-velocity coupling algorithm for all computational trials in this research and the flow domain is resolved by the pressure-based solver in a segregated manner.

The general function and implementation of the SIMPLE algorithm within the FLUENT CFD software is congruent with the extensive discussion in Section 2.1.2 and redundant consideration is unwarranted. However, as a reference to the form of equations described within the FLUENT software above, the fundamental equations used in the SIMPLE algorithm are accordingly reformulated for completeness to the current elucidation. As previously outlined in Section 2.1.2, the SIMPLE solution procedure applies a relationship between velocity and pressure corrections in order to enforce the mass conservation constraint and attain the corresponding pressure field. The equivalent form of the velocity correction (or flux correction) in FLUENT is expressed as (FLUENT Inc., 2006)

$$J_f' = d_f(p_{c0}' - p_{c1}') \tag{2.91}$$

where, (p') is the cell pressure correction. A discrete equation for the cell pressure correction (p') is obtained from Equations (2.91) and (2.88), which is given by (FLUENT Inc., 2006)

$$a_P p' = \sum_{nb} a_{nb} p_{nb}' + \sum_{f}^{N_{faces}} J_f^* A_f \tag{2.92}$$

where, (J_f^*) is the face flux resulting from the initially guessed pressure field (p^*) provided at the start. Subsequently, a solution is obtained for (p') in Equation (2.92) and the cell pressure as well as face flux are corrected respectively as follows (FLUENT Inc., 2006)

$$p = p^* + \alpha_p p' \tag{2.93}$$

$$J_f = J_f^* + d_f(p'_{c0} - p'_{c1}) \tag{2.94}$$

where, (α_p) is the under-relaxation factor for pressure. The iterative procedure continues until the mass conservation constraint is sufficiently realized.

- **Steady-State Under-Relaxation:** In steady-state flow calculations for the pressure-based solver, the governing equations do not include any time-dependent terms or explicit time steps upon which consecutive iterations for updating variable values (*i.e.* ϕ) are controlled. Due to the nonlinearity of the equation set evaluated, the under-relaxation of variables is essential for controlling the change of (ϕ) by reducing the variance in (ϕ) values produced during successive iterations. Simply stated, the new cell value for a variable (ϕ) is determined according to the following expression (FLUENT Inc., 2006)

$$\phi = \phi_{old} + \alpha\Delta\phi \tag{2.95}$$

where, (ϕ_{old}) is the previous iteration value, (α) is the under-relaxation factor, and ($\Delta\phi$) is the calculated change in the (ϕ) value. In addition, the under-

relaxation of equations is also necessary in order to alleviate instabilities associated with the nonlinear outer loop iterations. This is achieved by regulating the amounts of (ϕ) introduced into the discretized system of equations and is somewhat akin to the time step used in the temporal discretization of time-dependent equations. The implicit relaxation of equations is defined as (FLUENT Inc., 2006)

$$\frac{a_P \phi}{\alpha} = \sum_{nb} a_{nb} \phi_{nb} + b + \left(\frac{1-\alpha}{\alpha}\right) a_P \phi_{old} \tag{2.96}$$

where, (α) is again the under-relaxation factor and (b) is the appropriately related source term for the cell.

A general solution algorithm representing the pressure-based segregated flow solver in the FLUENT CFD software is illustrated in Figure 2.11 as a reference to the basic solver implementation outlined in the above discussion. This solver formulation along with the SIMPLE pressure-velocity coupling algorithm is exclusively employed in all computational experiments conducted in this research.

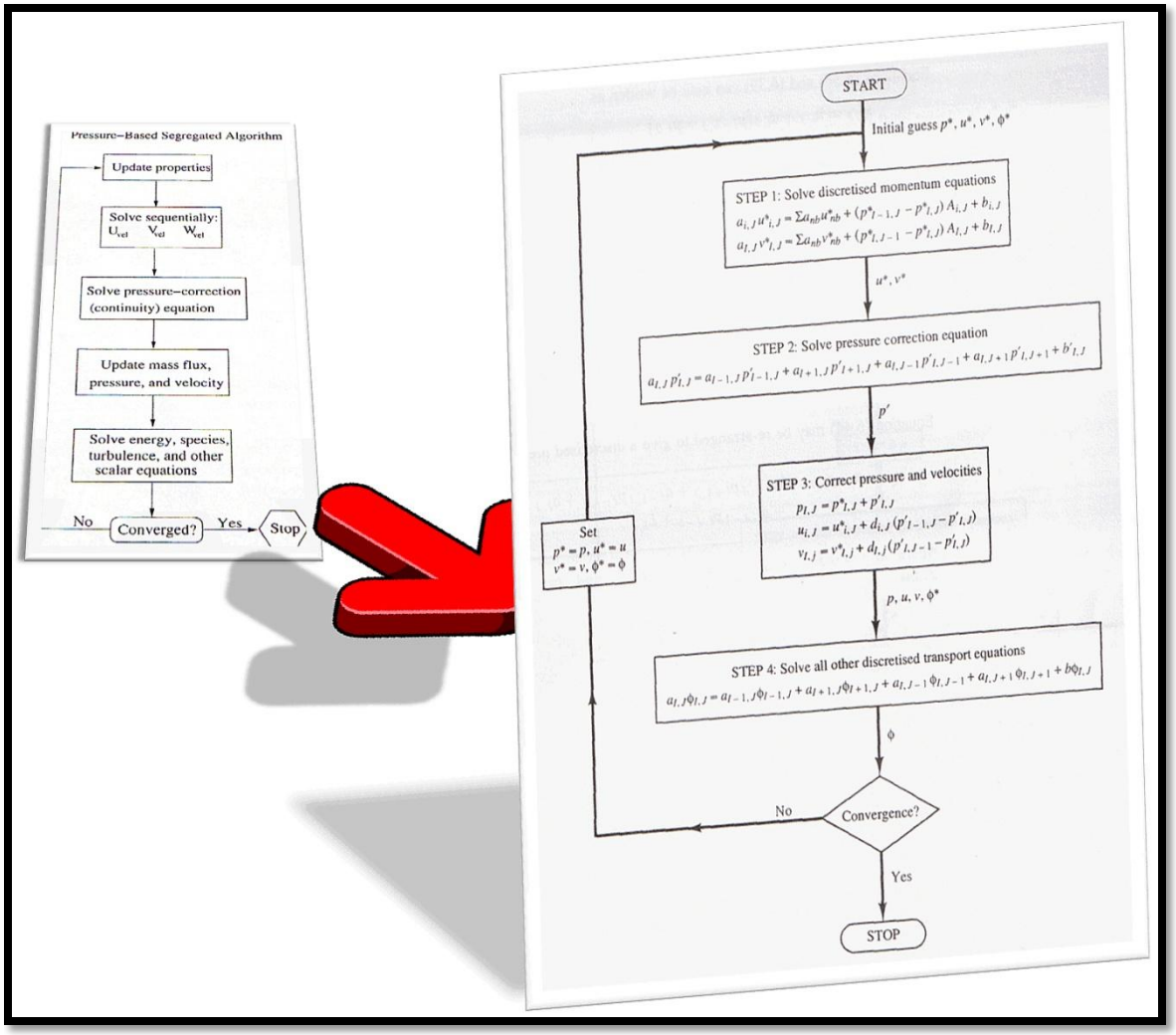


Figure 2.11: General outline of the pressure-based segregated solver algorithm [back] and sequence of operations in a typical CFD procedure employing the SIMPLE algorithm [front]; adapted in part from (FLUENT Inc., 2006; Versteeg and Malalasekera, 1995).

2.3 SOLUTION PROCEDURE

The foremost issue of concern with regard to any solution methodology is the necessary development of an organized, structured outline describing a generalized procedure that, when adhered to properly, attains the pre-designated output sought in a

continually repeatable fashion. There are six fundamental steps summarizing the solution procedure followed for conducting the computational experiments in this research: (1) obtain physical fuel assembly dimensions required for model; (2) create CAD solid model of fuel assembly components modeled within the flow domain; (3) use solid model to construct flow domain in appropriate preprocessor software; (4) mesh computational flow domain using preprocessor software; (5) setup flow solver and calculate flow field solution on the mesh model; and (6) postprocess solution results for evaluation. This sequence of tasks forming the general solution procedure for a computational experiment involving a low-Re/transitional viscous flow is graphically illustrated in Figure 2.12 below, which includes a few key statements of consideration associated with each step of the solution process.

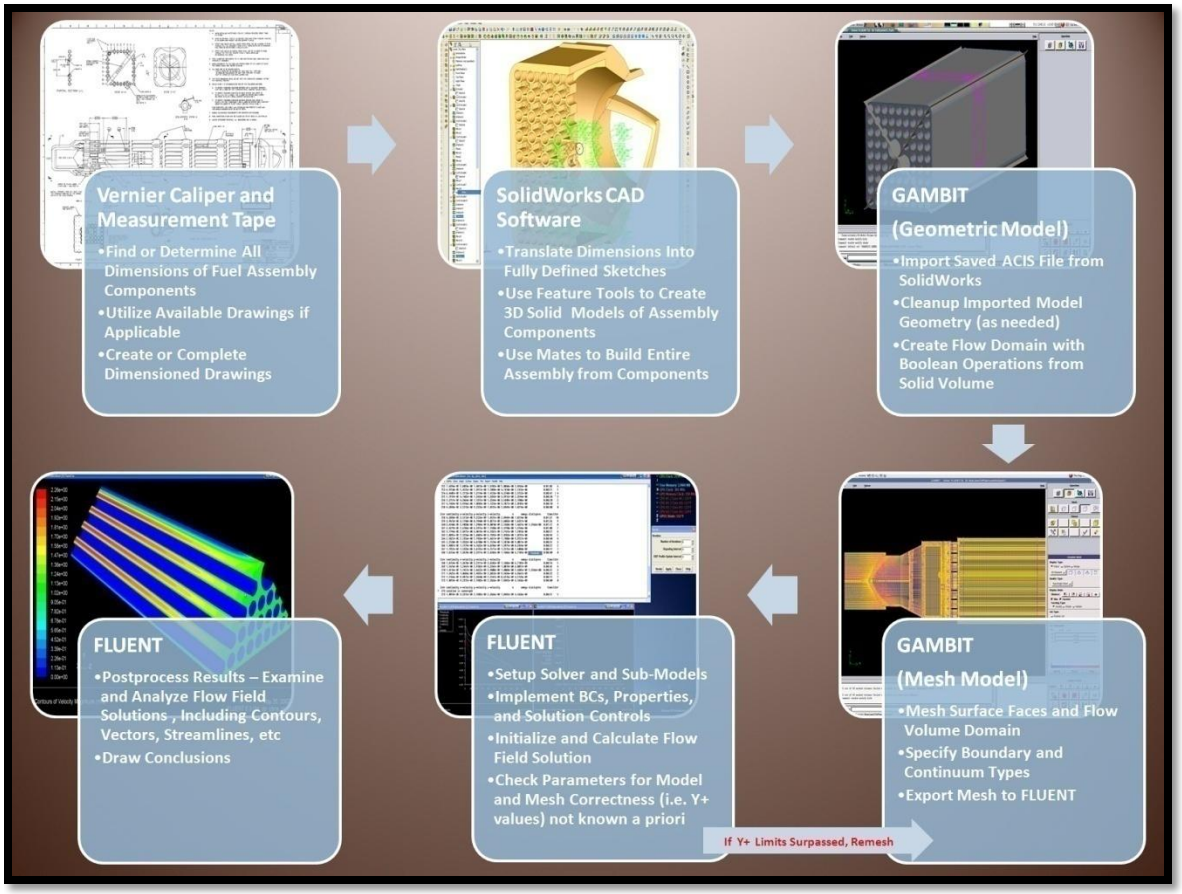


Figure 2.12: Outline of general solution procedure for a computational experiment involving a low-Re/transitional viscous flow.

A number of the computational modeling tools utilized in the solution procedure are included within the FLUENT CFD software package, which is comprised of several different program entities including: (1) CAD/CAE filters that function as translators for importing solid surface geometries of basic output file format from other CAD software; (2) geometry and mesh building intelligent toolkit (GAMBIT) preprocessor for constructing model geometry and mesh generation; and (3) FLUENT flow solver for performing computational calculations on the mesh domain and postprocessing solution results. Steps (1) through (4) of the solution procedure presented in Figure 2.12 are

relatively straightforward in their description and specific details surrounding the construction of fuel assembly models incorporated into this research are discussed in Chapter 4 of this dissertation. The final two tasks, however, encompass a subroutine of additional guideline steps followed for each computational experiment and necessitate further elaboration. Although the precise characteristics of the process are largely problem dependent, a generalized outline that sufficiently identifies the primary considerations taken into account for a number of flow analyses is provided as a reference in Figure 2.13 below. The solution approach and particular analysis details associated with individual computational experiments are imparted accordingly in Chapter 5 discussions attributed to each experimental trial.

BASIC FLUENT CFD SOLUTION STEPS	
1. Launch Appropriate Solver	Select 2D or 3D, Precision, Parallel or Serial, Check Spawned Nodes…
2. Import Mesh	Partition Grid (if needed), Check Grid for Error, Scale and Change Units (if needed), Check Partition Statistics, Setup / Adapt Grid…
3. Define Solver Formulation	Pressure or Density Based, Implicit / Explicit, Steady or Unsteady…
4. Governing Equations / Sub-models Activated	Laminar or Turbulent Flow, Heat Transfer Models, Radiation, Energy, Porous Media…
5. Specify Material Properties	Select Existing Fluid / Solid Material Database, or Enter New Values
6. Define Operating Conditions	Gravity Magnitude and Axis Direction Applied
7. Enter Boundary Conditions	Set / Modify Boundary Types or Enter Boundary Values for Temperature, Adiabatic, Radiation, Convection, Periodic Input…
8. Adjust Solution Control Parameters	Choose Equations Solved, Setup any Monitors, Under-Relaxation Factors, P-V Scheme, Discretization Upwind…
9. Initialize Flow Field	Select Parameter Values to Drive Initial Conditions of Domain
10. Set AMG Controls	Enter AMG Scalar Parameters, Cycle Type, Options…
11. Iterate / Post-process	Set Time Steps / Number Iterations, Calculate a Solution, Post-process Results, Save Case / Data
12. Refine Grid / Revise Case Setup	If Y^+ Values Exceed Limit or Solution is Inconsistent, Refine / Regenerate Mesh

Figure 2.13: Example of basic solution steps followed within FLUENT CFD software for several computational experiments.

2.4 NUMERICAL ACCURACY

In the last twenty-five years, one of the fastest growing concerns with regard to the widespread use of modern CFD analysis is the absence of a globally standard policy or criteria requiring engineering authors to maintain a level of professional stature in reporting the numerical solution of a computational fluids analysis. In order to promote confidence in the accuracy and legitimacy of numerical experimentation within the computational science community, efforts to assess the numerical uncertainty and accuracy, explain the basic governing physics modeled, justify the numerical techniques employed, and provide sufficient clarity of detail upon which replication of the results is possible, must be commonplace among the published literature. As a result, the individual efforts of several well-known professional engineering journals have culminated in the adoption of policy statements on the control of numerical accuracy with the purpose of establishing acceptable standards for publication within their respective media (Freitas, 1993; Roache *et al.*, 1986; The Editorial Board, n.d.). One such early policy example states the following:

> The Journal of Fluids Engineering will not accept for publication any paper reporting the numerical solution of a fluids engineering problem that fails to address the task of systematic truncation error testing and accuracy estimation (Roache *et al.*, 1986).

Therefore, to put forth the 'best-effort' of the author to adequately provide accurate, validated numerical results based on sound computational research, a set of appropriate criteria for assessing numerical uncertainty as outlined in the ASME Journal of Fluids Engineering (Freitas, 1993) is adopted as a guideline for conducting and

presenting solutions of the computational experiments throughout this research. For reference, the summarized criteria statements adapted from the ASME Journal of Fluids Engineering for the assessment of numerical uncertainty are listed as follows:

1. Basic solution method details with information regarding truncation error of individual terms in the governing equations must be described.
2. Second-order accurate methods or higher must be employed for spatial discretization.
3. Inherent (or explicit) artificial viscosity/diffusivity must be minimized.
4. Iterative convergence criteria and considerations must be addressed.
5. Grid independence (or convergence) must be established.
6. Transient calculations must assess and minimize the amount of phase error.
7. The accuracy and appropriateness of boundary conditions as well as initial flow field values implemented must be explained.
8. Existing software code must be fully cited in widely available references.
9. Solutions from benchmark studies may be employed for validation of results for a specific class of problems.
10. In addition, reliable experimental results may be utilized to validate a set of simulated solutions.

According to Freitas (1993), it is reasonable to require papers on CFD obtained solutions to realize basic criteria as outlined above before considered appropriate for presentation within the respective journal. As previously discussed in Section 1.5, despite the expenditure of substantial effort throughout this research to address each of the criteria assessments listed, it is reasonably understood that several modeling tradeoffs

in the form of limited fulfillment of the criteria in a few simulation trials is necessary in order to provide a more-complete, meaningful scope of research. Specifically, consideration of the impractical effort required for complete model validation of the large-scale simulations precludes a substantial validation analysis outside of the models used in preliminary scoping studies. The results of these initial case studies and model validations are assumed as accurate reflections of the greater fuel assembly domain where one or more of the assessment criteria listed above is ineffectively addressed on a large-scale set of simulations. Numerical uncertainty is discussed in detail with each computational experiment set presented in Chapter 5 of this dissertation.

Chapter 3: Parallel Physical Experiments Conducted by SNL

3.1 INTRODUCTION

The primary intention of this chapter is to provide a brief summary of the SFP pressure drop experiments conducted by SNL and present a limited selection of the hydraulic analysis results, which are fundamental for a comprehensive understanding of the parallel computational experiments elucidated in Chapter 5 deliberations. As discussed throughout Chapter 1, heightened concern with regard to the safety of SFPs under complete LOCA scenarios provoked efforts to extend the modeling capabilities of the MELCOR severe accident analysis code to include SFP configurations and accident phenomena, which culminated in the MELCOR SFP model. Intrinsically, the MELCOR code (and inherent MELCOR SFP model) relies heavily on constitutive or phenomenological relationships to quantify the physics influencing typical system response during accident progressions.

The hydraulic constitutive relationships of particular importance in complete loss-of-coolant accidents specify the viscous (or major) and inertial (or minor) losses along a flow path as a hydraulic flow loss term to the momentum equation (Durbin and Lindgren, 2005). These flow losses work against the buoyancy-driven convection induced by temperature gradients and are vital precursors to the type of transient thermal behavior and nature of heat propagation reaction. Ultimately, the effectiveness of the heat removal rate from a fuel assembly is unequivocally reliant upon the overall balance between developing buoyancy forces and total flow resistance (viscous and inertial losses) across the assembly flow domain. Therefore, the predicted fuel assembly response (and corresponding accuracy) is exceptionally sensitive to the validity of the hydraulic loss

coefficient inputs required for the major and minor flow loss constitutive relationships. Initial scoping evaluations conducted by SNL utilizing a pre-calibration (*i.e.* prior to SFP pressure loss experiments) version of the MELCOR SFP model confirmed this conclusion (Gauntt and Wagner, 2004).

Unfortunately, experimental data on the hydraulic and thermal response of a typical fuel assembly heating up in air is virtually nonexistent (Lindgren, 2004). Consequently, a need is presented to obtain basic thermal hydraulic data for a SFP fuel assembly under complete loss-of-coolant accident conditions to facilitate and/or validate flow resistance modeling and calibration of the MELCOR SFP model. This is the primary motivation for SNL conducting the SFP pressure drop experiments and necessitating experimentally verified results to reduce concerns associated with the interpretation of model-based formulations. To succinctly sum up, the formally stated purpose of the experimental investigations and subsequent hydraulic analysis of data discussed in this chapter is to determine the hydraulic loss coefficients, namely (S_{LAM}) and (k) values, for calibration of the MELCOR severe accident analysis code and SFP model (Durbin and Lindgren, 2005). Finally, it is with great emphasis and a clear understanding that SNL is acknowledged as the sole source of all experimental efforts and results discussed/presented in this chapter; and only through the willingness of SNL and the NRC (agency providing the funding for the stated research) to disclose this information is the following discussion possible.

3.2 EXPERIMENTAL SETUP

The experimental apparatus consists of a highly prototypic fuel assembly derived from the Global Nuclear Fuel (GNF) 9x9 BWR (GE 11 BWR/2-3) design. Commercially

available components purchased by SNL constitute the bulk of hardware used in constructing the prototypic experimental fuel assembly including the lower tie plate (LTP), upper tie plate (UTP), spacers, water rods, and channel box (or canister). However, the fuel rod pins incorporated into the experimental assembly are slightly modified from the actual fuel rod design and stainless steel conduit is substituted for the fuel rod material. These simplified mock fuel pins are fabricated according to dimensioned drawings and physical examples obtained from GNF, but with the exception of a marginally smaller diameter. The basic dimensions of the prototypic fuel assembly components are provided in Table 3.1 below.

Table 3.1: Basic dimensions in meters for fuel assembly components reported in the hydraulic analysis studies by SNL (Durbin and Lindgren, 2005).

Description	Lower (Full) Section	Upper (Partial) Section
Number of pins	74	66
Pin diameter (m)	1.11×10^{-2}	1.11×10^{-2}
Pin pitch (m)	1.44×10^{-2}	1.44×10^{-2}
Pin separation (m)	3.28×10^{-3}	3.28×10^{-3}
Water rod OD [main section] (m)	2.49×10^{-2}	2.49×10^{-2}
Water rod ID (m)	2.34×10^{-2}	2.34×10^{-2}

In order to determine pressure losses across both single and multiple component flow segments of the fuel assembly, pressure ports are strategically placed through the canister wall at varying height locations along the axial length of the fuel assembly. A diagram of the experimental apparatus for the SFP pressure drop experiments including all pressure port locations is illustrated in Figure 3.1 below. Two Paroscientific Digiquartz differential pressure transducers (Model 1000-3D), which utilize an extremely sensitive quartz crystal to measure slight variations in differential pressure (capable resolution of approximately 0.02 Pa), are employed in attaining the pressure loss

measurements and are plumbed directly to the desired pressure ports (Durbin and Lindgren, 2005). The nomenclature used by SNL in reporting the pressure drop associated with a specific flow segment traversed by two pressure ports is indicated according to the downstream followed by upstream port numbers. For example, the pressure drop across the bundle run and grid spacer segment immediately downstream of the lower tie plate in Figure 3.1 is expressed as the segment (15-17) pressure drop.

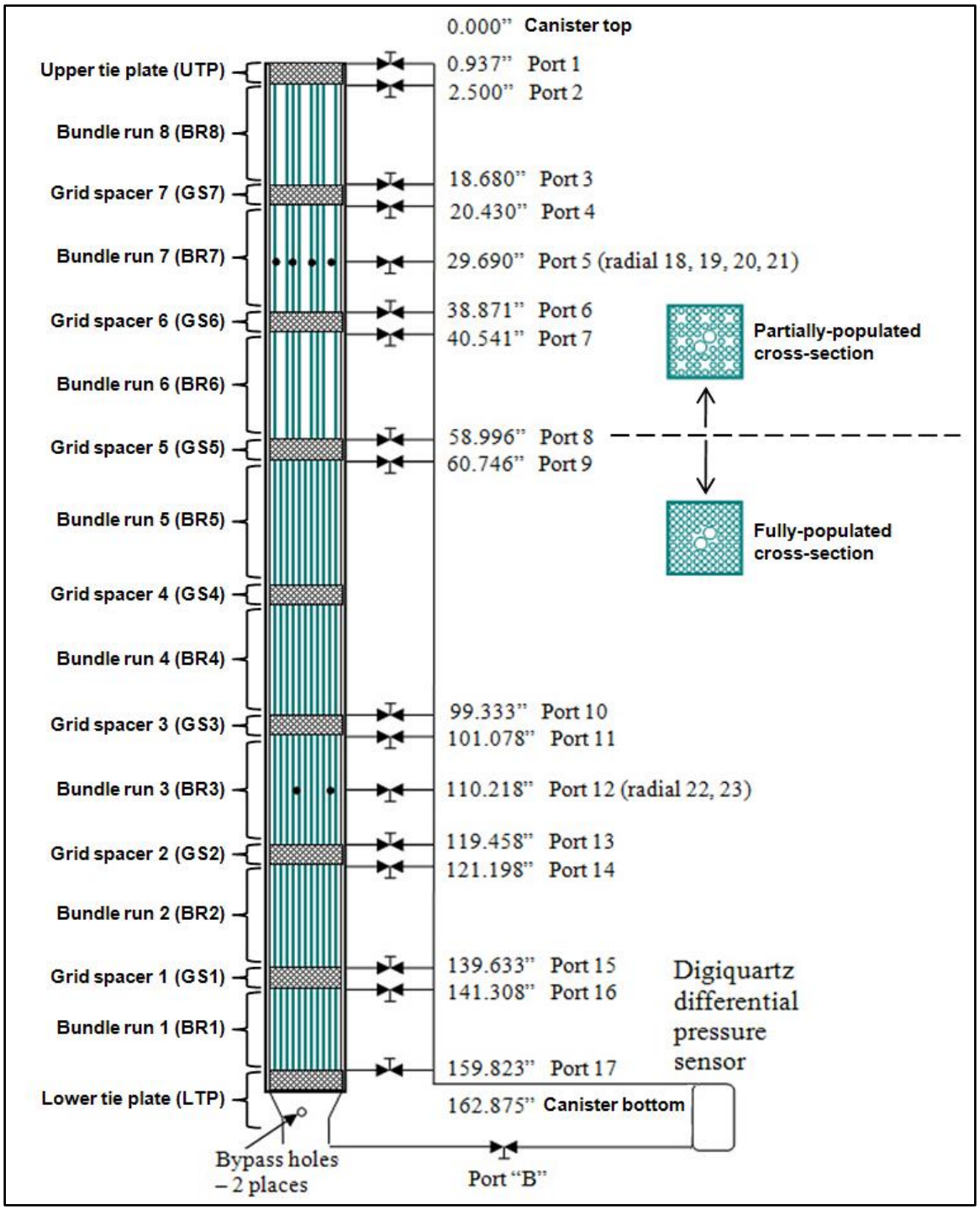


Figure 3.1: Diagram of SNL experimental setup with pressure port locations; adapted in part from (Durbin and Lindgren, 2005).

During each experimental trial, the measurements are recorded directly to the hard disk of a PC-based data acquisition system at three-second intervals utilizing a National Instruments LabVIEW® 7.1 software interface (Durbin and Lindgren, 2005). The experimental measurements include the airflow rate, ambient air pressure and temperature, and flow segment pressure drops recorded over a two-minute period for each case evaluated. Transducer zero-drift corrections and initial pressure spikes at the onset of establishing the evaluated airflow rates are all properly accounted for in determining the final experimental results. Additional information with regard to the data collection procedure followed in the SFP pressure drop experiments is available in the hydraulic analysis report summary by SNL (Durbin and Lindgren, 2005).

On a final note, results from the group of physical experimental runs directly compared to the computational experiments in this research are for the 'blocked water rods' apparatus only. In this series of experiments, interest is focused on the pressure losses across fuel assembly flow segments in which all of the airflow is forced through the fuel bundle and spacers encompassed by the canister (*i.e.* zero flow through water rods or bypass holes in LTP). This is achieved by inserting an acrylic sleeve over the water rod outlet holes located at the downstream extent of each water rod and covering both bypass holes in the LTP. The cross-sectional flow areas and hydraulic diameters reported by SNL for both the standard and 'blocked water rods' experiment setups are compared in Table 3.2 below.

Table 3.2: Summary of cross-sectional flow areas and hydraulic diameters of the SFP experiments reported by SNL in the hydraulic analysis studies (Durbin and Lindgren, 2005).

Segment Description	A_c (m^2)	D_H (m)
Upper (Partially-Populated) Bundle	1.06×10^{-2}	1.41×10^{-2}
Upper (Partially-Populated) Bundle + Water Rods	1.14×10^{-2}	1.45×10^{-2}
Upper (Partially-Populated) Spacer	8.35×10^{-3}	3.57×10^{-3}
Lower (Fully-Populated) Bundle	9.79×10^{-3}	1.19×10^{-2}
Lower (Fully-Populated) Bundle + Water Rods	1.06×10^{-2}	1.24×10^{-2}
Lower (Fully-Populated) Spacer	7.37×10^{-3}	2.89×10^{-3}
Water Rod (×1)	4.29×10^{-4}	2.49×10^{-2}

3.3 HYDRAULIC ANALYSIS

In order to determine the hydraulic loss coefficients (S_{LAM}) and (k) for calibrating the MELCOR severe accident analysis code and SFP model, curve fits to the experimentally measured pressure drop data are formulated based on constitutive relationship forms for both major and minor flow losses as inherently expressed within the program code. As previously discussed, a distinct equation for each of the major and minor pressure losses is defined such that the total pressure loss is given by

$$\Delta P_{total} = \Delta P_{major} + \Delta P_{minor} \tag{3.1}$$

The major (or viscous) pressure loss in Equation (3.1) is expressed based on a form of the Darcy-Weisbach phenomenological equation and is written as (Durbin and Lindgren, 2005)

$$\Delta P_{major} = f\left(\frac{L}{D_H}\right)\left(\frac{\rho V_{bundle}^2}{2}\right) \tag{3.2}$$

where, (f) is the friction factor, (L) is the flow segment length, (D_H) is the hydraulic diameter, and (V_{bundle}) is the average fluid velocity in the fuel bundle (*i.e.* bundle run). Friction factors (f) are stated explicitly for a limited number of cases involving fully developed laminar flow, which include the following examples:

$$f = \frac{64}{Re} \quad \text{[pipe flow]} \tag{3.3}$$

$$f = \frac{100}{Re} \quad \text{[bundle of cylinders]} \tag{3.4}$$

The friction factor relationship is determined from an analytical solution for circular pipe flow (Fox and McDonald, 1992) in Equation (3.3) and flow along cylinders arranged in a square array (Sparrow and Loeffler, 1959) in Equation (3.4). In the hydraulic analysis of the experimental pressure drop data, (f) in Equation (3.2) is defined as (Durbin and Lindgren, 2005)

$$f = \frac{S_{LAM}}{Re} \tag{3.5}$$

and, the Reynolds number (Re) is given by

$$Re = \frac{\rho V_{bundle} D_H}{\mu} \tag{3.6}$$

Rewriting Equation (3.5) using the expression for (Re) in Equation (3.6), and substituting into Equation (3.2) yields

$$\Delta P_{major} = S_{LAM}\left(\frac{L}{D_H^2}\right)\left(\frac{V_{bundle}\mu}{2}\right) \tag{3.7}$$

Therefore, the final form of the major pressure loss in Equation (3.1) is given by Equation (3.7) above.

The minor (or form) pressure loss in Equation (3.1) is simply defined in the hydraulic analysis of the experimental pressure drop data as (Durbin and Lindgren, 2005)

$$\Delta P_{minor} = \sum k\left(\frac{\rho V_{bundle}^2}{2}\right) \tag{3.8}$$

Substituting Equations (3.7) and (3.8) into Equation (3.1), the expression for the total pressure loss becomes

$$\Delta P_{total} = S_{LAM}\left(\frac{L}{D_H^2}\right)\left(\frac{V_{bundle}\mu}{2}\right) + \sum k\left(\frac{\rho V_{bundle}^2}{2}\right) \tag{3.9}$$

Combining constants within each group of terms representing the major and minor pressure losses, Equation (3.9) is recast in the following form (Durbin and Lindgren, 2005)

$$\Delta P_{total} = a_2 V_{bundle}^2 + a_1 V_{bundle} \tag{3.10}$$

Since the (a_1) and (a_2) terms are constants, curve fits to the experimentally measured pressure drop data are presented in the format of Equation (3.10) for flow segment

pressure loss versus average fluid velocity. The values of (a_1) and (a_2) are subsequently determined by the coefficients of the polynomial curve fits, which permit the explicit calculation of sought (S_{LAM}) and (k) coefficient values from their representation in Equation (3.10) as follows:

$$S_{LAM} = 2a_1\left(\frac{D_H^2}{L\mu}\right) \tag{3.11}$$

$$\sum k = \frac{2a_2}{\rho} \tag{3.12}$$

Results from a limited number of the curve-fit analyses conducted by SNL for the SFP pressure drop experiment trials are discussed in the following section of this chapter.

3.4 Results and Final Remarks

The SNL experimental apparatus is operated in the laminar flow regime with expected Reynolds numbers spanning 70 to 900, based on fuel bundle velocity and hydraulic diameter (Durbin and Lindgren, 2005). Curve fits to the pressure drop data across ports (2-8), (8-17), and (2-17) are illustrated in Figure 3.2 for a particular hydraulic analysis of the experimental fuel assembly with blocked water rods. The pressure port locations are shown in Figure 3.1 and the cross-sectional flow areas along with hydraulic diameters of each flow segment are provided in Table 3.2 above. Pressure losses in Figure 3.2 are plotted against corresponding velocities and geometry characteristics of the associated fully- (8-17) or partially-populated (2-8) bundle run sections. However, the velocities, cross-sectional flow area, and hydraulic diameter of the fully-populated bundle run section are used for the segment (2-17) data. In addition,

the (2-17) pressure drops are the summation of both the (2-8) and (8-17) section losses, and do not represent experimental measurements recorded (Durbin and Lindgren, 2005).

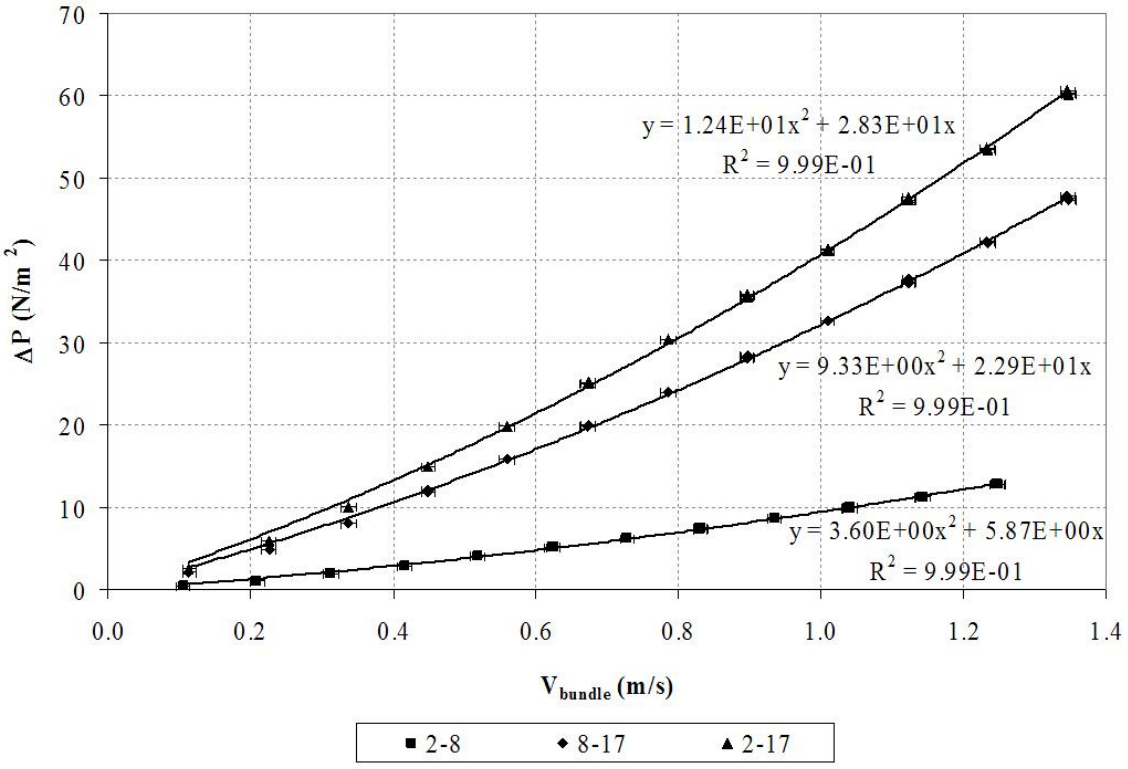


Figure 3.2: Curve fits to the pressure drop data as a function of corresponding bundle run velocity for the SNL experimental fuel assembly with blocked water rods (Durbin and Lindgren, 2005).

The resulting (S_{LAM}) and (k) coefficient analysis information pertaining to the curve-fit pressure drop data in Figure 3.2 is documented in Table 3.3 below. This hydraulic analysis assumes the properties of air reside at local ambient conditions, typically a density of ($\rho = 0.98$ kg/m^3) and dynamic viscosity of ($\mu = 1.85\text{x}10^{-5}$ N-s/m^2); although, differences in air pressure and temperature are accounted for in measurements recorded during other experimental trials (Durbin and Lindgren, 2005).

Table 3.3: Resulting coefficient analysis data from pressure loss curve fits in Figure 3.2 for segments (2-8), (8-17), and (2-17) with blocked water rods (Durbin and Lindgren, 2005).

Segment	L (m)	A_c (m^2)	D_H (m)	a_1 ($N \cdot s/m^3$)	a_2 ($N \cdot s^2/m^4$)	S_{LAM}	Σk	No. spacers	k per spacer
2-8	1.43	0.0106	0.0141	5.87	3.60	88	7.3	2	3.7
8-17	2.56	0.0098	0.0119	22.9	9.33	138	19	5	3.8
2-17	4.00	0.0098	0.0119	28.3	12.4	109	25	7	3.6

Uncertainties in all (S_{LAM}) and (k) coefficient values listed in the hydraulic analysis report by SNL are stated as (±5) and (±1), respectively (Durbin and Lindgren, 2005).

The experiments also considered the influence of variations to the airflow rate range included in the hydraulic analysis. In this study, subsets of the entire data set for the (2-8), (8-17), and (2-17) segment pressure losses are used to calculate the curve-fit coefficients. The results of this study for varying flow rate ranges in standard liters per minute (slpm) are provided in Table 3.4 below.

Table 3.4: Curve-fit coefficient values determined for different ranges of experimental airflow rate (Durbin and Lindgren, 2005).

Flow Rate Range (slpm)	2-8		8-17		2-17	
	S_{LAM}	Σk	S_{LAM}	Σk	S_{LAM}	Σk
50-200	46	20	99	47	75	64
50-300	60	14	113	34	86	46
50-400	75	10	126	25	99	33
50-500	84	8.1	134	20	106	27
50-600	88	7.3	138	19	109	25

There are rather significant variances in the (S_{LAM}) and (k) coefficient values for varying ranges of flow rate as shown in Table 3.4, which is a potentially troublesome situation depending upon the accuracy with which airflow rates expected to develop are known *a priori* or at least demonstrated within reasonable constraints to have minimal effect on the results. This is especially true for complete LOCA conditions in which buoyancy-

induced airflows serve as a primary heat removal mechanism for SFP assemblies. Nevertheless, the majority of all hydraulic loss coefficient values reported by Durbin and Lindgren (2005) are calculated based on the full experimental flow rate range of 50 to 600 slpm.

A summary of the (S_{LAM}) and (k) coefficients determined from the SFP pressure drop experiments with blocked water rods is presented in Table 3.5 below for the full experimental flow rate range examined. The cross-sectional area of flow and hydraulic diameter of the fully-populated bundle run section are used to calculate the coefficient values for any flow segment span that includes the lower (fully-populated) region of the fuel assembly [*i.e.* (1-B)]. However, coefficient values for flow segments (1-2), (4-6), and (6-7) are computed according to the hydraulic characteristics of both the fully- and partially-populated bundle run sections (Durbin and Lindgren, 2005).

Table 3.5: Summary of coefficient values for experiments with blocked water rods over the full experimental flow rate range (Durbin and Lindgren, 2005).

Description	Segment	A_c (m^2)	L (m)	D_H (m)	S_{LAM}	Σ k	No. spacers	k per spacer
Long Spans	1-B	0.0098	4.19	0.0119	106	37	–	–
	2-8	0.0106	1.44	0.0141	88	7.3	2	3.6
	8-17	0.0098	2.56	0.0119	138	19	5	3.8
	2-17	0.0098	4.00	0.0119	109	25	7	3.6
	4-8	0.0106	0.98	0.0141	85	4.0	1	4.0
	8-15	0.0098	2.05	0.0119	137	15	4	3.8
	4-15	0.0098	3.03	0.0119	111	18	5	3.6
Individual Bundle Runs	4-5	0.0106	0.24	0.0141	69	0.88	–	–
	5-6	0.0106	0.23	0.0141	51	0.00	–	–
	4-6	0.0106	0.47	0.0141	60	0.88	–	–
	4-6	0.0098	0.47	0.0119	40	0.71	–	–
	11-12	0.0098	0.23	0.0119	89	0.70	–	–
	12-13	0.0098	0.23	0.0119	65	0.13	–	–
	11-13	0.0098	0.47	0.0119	77	0.84	–	–
Spacers / Tie Plates	1-2	0.0106	0.040	0.0141	49	0.42	–	–
	1-2	0.0098	0.040	0.0119	33	0.36	–	–
	6-7	0.0106	0.045	0.0141	408	3.1	1	3.1
	6-7	0.0098	0.045	0.0119	272	2.6	1	2.6
	13-14	0.0098	0.044	0.0119	729	3.2	1	3.2
	17-B	0.0098	0.13	0.0119	119	11	–	–

The (S_{LAM}) and (k) coefficients are also determined for spacer segments (6-7) and (13-14) using the actual cross-sectional area of flow and hydraulic diameter appropriate to each spacer component. These values are included in Table 3.6 below.

Table 3.6: Coefficients calculated for spacer segments using the actual cross-sectional area of flow and hydraulic diameter of each spacer (Durbin and Lindgren, 2005).

Segment	A_c (m^2)	L (m)	D_H (m)	S_{LAM}	Σ k
6-7	0.00835	0.045	0.00357	21	1.9
13-14	0.00737	0.044	0.00289	32	1.8

Furthermore, Durbin and Lindgren (2005) postulate that (S_{LAM}) and (k) coefficients observed in Table 3.5 for assembly spans encompassing a multiple component segment (*i.e.* 8-15) are obtainable by appropriately weighting the (S_{LAM}) and (k) values from Table 3.5 for each single component segment (*i.e.* 13-14) of unique geometry comprising the multiple component span considered. Specifically, the (S_{LAM}) and (k) coefficient values from relevant single component segments are individually weight averaged based on the flow length and number of instances the single component appears in the multiple component span, respectively. A comparison between the (S_{LAM}) and (k) coefficient values derived using the postulated calculations outlined above and those from direct experimental measurement is provided in Table 3.7 below for several multiple component segments. The notable discrepancies between these two sets of effective (calculated) and measured coefficient values reported by Durbin and Lindgren (2005) are addressed in Section 5.2 alongside the coefficient results determined from simulation trials involving the fuel bundle run and spacer domain model.

Table 3.7: Calculated effective and measured coefficient values compared for different span lengths encompassing a multiple component segment (Durbin and Lindgren, 2005).

Pressure Drop Segment	Measured		Effective	
	S_{LAM}	Σ k	S_{LAM}	Σ k
1-B	106	37	104	39
2-8	88	7.3	81	8.8
8-17	138	19	133	20
2-17	109	25	105	28
4-8	85	4.0	75	4.9
8-15	137	15	133	16
4-15	111	18	106	20

In conclusion, the cumulative pressure loss across the fuel assembly as a function of axial height location is illustrated in Figure 3.3 for six different airflow rates evaluated. The solid lines graphed in Figure 3.3 represent the summation of major and minor pressure losses determined by Equations (3.7) and (3.8), respectively, using the hydraulic loss coefficient values from Table 3.5 above. Delineation between the fully- and partially-populated regions of the fuel assembly is indicated by the dashed line at (z) equal to 2.7 meters.

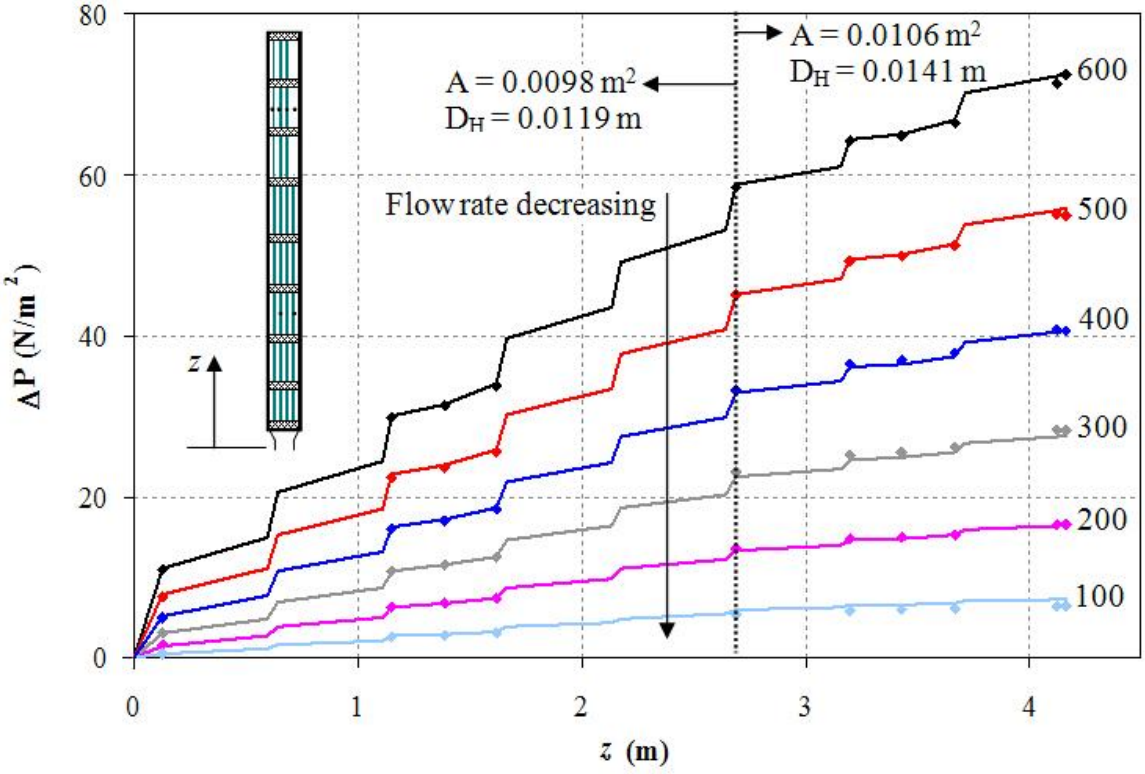


Figure 3.3: Cumulative pressure loss as a function of axial height location for six different flow rates at intervals spanning the full experimental flow rate range. The solid lines represent analytically determined pressure losses using coefficients from Table 3.5 (Durbin and Lindgren, 2005).

Chapter 4: Fuel Assembly Model Construction

4.1 CAD Solid Model

In order to facilitate rapid and accurate construction of flow domains for mesh generation among CFD models created, a complete CAD solid model of Sandia National Laboratories' prototypic GNF 9x9 BWR (GE 11 BWR/2-3) experimental fuel assembly is constructed using SolidWorks 2005, commercial 3D CAD software (SolidWorks Corporation, 2005). In most CFD modeling exertions, the physical domain of interest is established from previously created design/manufacturing drawings that include dimensions and tolerances utilized to create each component within the flow field. Otherwise, the required dimensions are typically obtained by physically taking the measurements by hand; using an instrument such as a digital vernier caliper is preferred for greater accuracy, but simple flexible measuring tape is often substituted in cases where digital instruments are not applicable for the necessary measurement(s).

Unfortunately, due to the proprietary nature of the fuel assembly design, a complete set of dimensioned drawings from the manufacturer is not attainable, although several completed component drawings obtained by SNL are available for reference. Therefore, remaining dimensions are individually measured from components already purchased/fabricated by SNL in order to complete the required drawings. The tolerance specification of the digital vernier caliper used to acquire absent component dimensions is stated by the manufacturer as ± 0.01 mm. Inherent human measurement tolerance is assumed roughly equivalent to ± 0.25 mm for larger measurements over 10 cm in length. As a result, the overall uncertainty in these measurements is conservatively estimated by

simply adopting the vernier caliper instrument tolerance plus the reasonable average margin for human error, which results in a total uncertainty of approximately ± 0.26 mm.

The completed 3D solid 'assembly' file is compiled in a bottom-up approach from mated solid 'part' files representing each individual component comprising the entire fuel assembly (SolidWorks Corporation, 1995-2002). This first and only CAD solid model of the experimental assembly serves as an invaluable future reference and widely accessible resource to SNL researchers since future high-temperature experiments proposed are expected to result in the complete destruction of the prototypic experimental fuel assembly (Lindgren, 2004). Due to the significant number of assembly components and the large geometric aspect ratios associated with a majority of these constituents, only a select few images of the more predominant individual solid model parts are included below with an accompaniment of figures illustrating the complete fuel assembly CAD model. An example scale of the detail incorporated into each component model is apparent by noting geometrical characteristics of the plug CAD models for each fuel rod type in Figures 4.1 and 4.2 compared with the size of scale for assembly part models in Figures 4.3 through 4.9 and the entire fuel assembly in Figures 4.10 through 4.12 below.

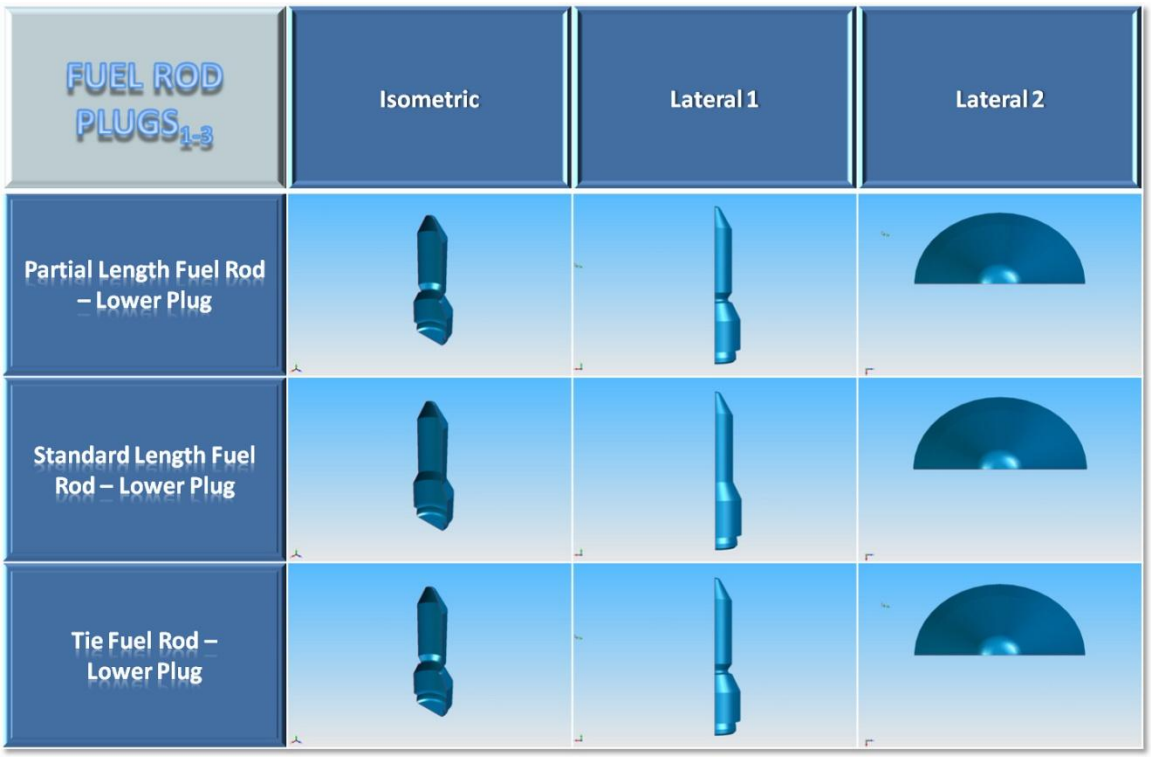


Figure 4.1: CAD solid models of lower plugs for all fuel rod types.

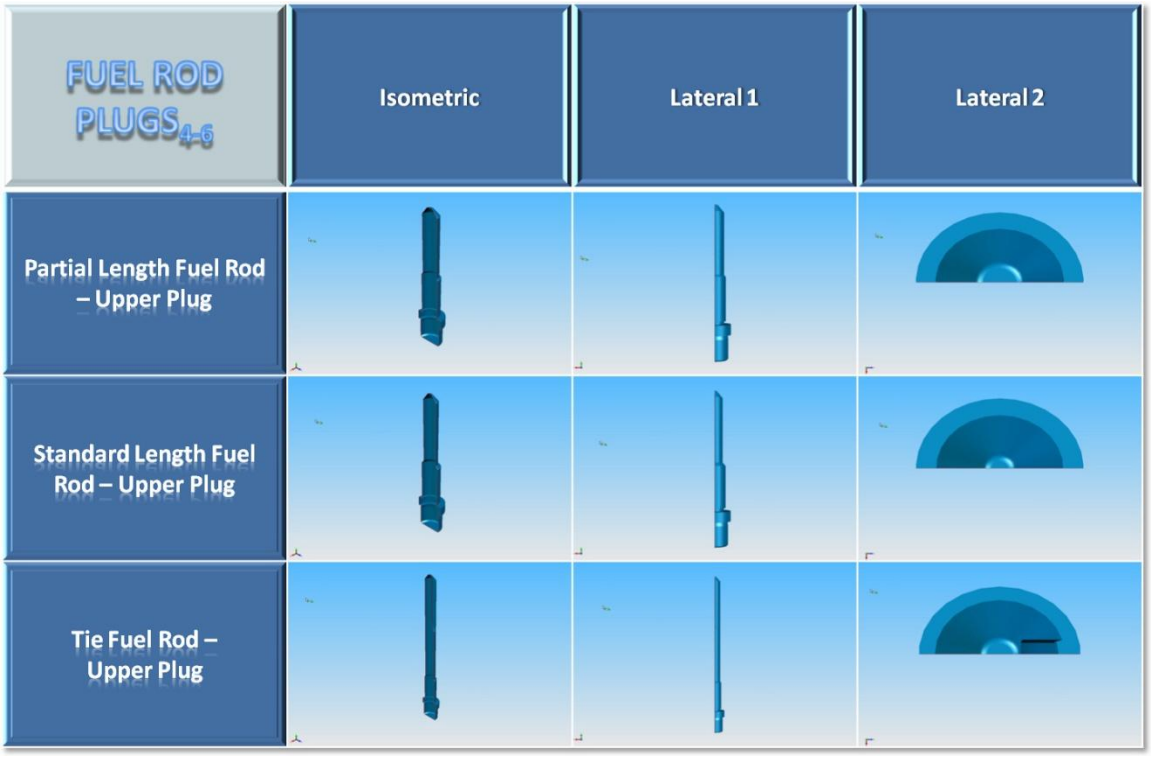


Figure 4.2: CAD solid models of upper plugs for all fuel rod types.

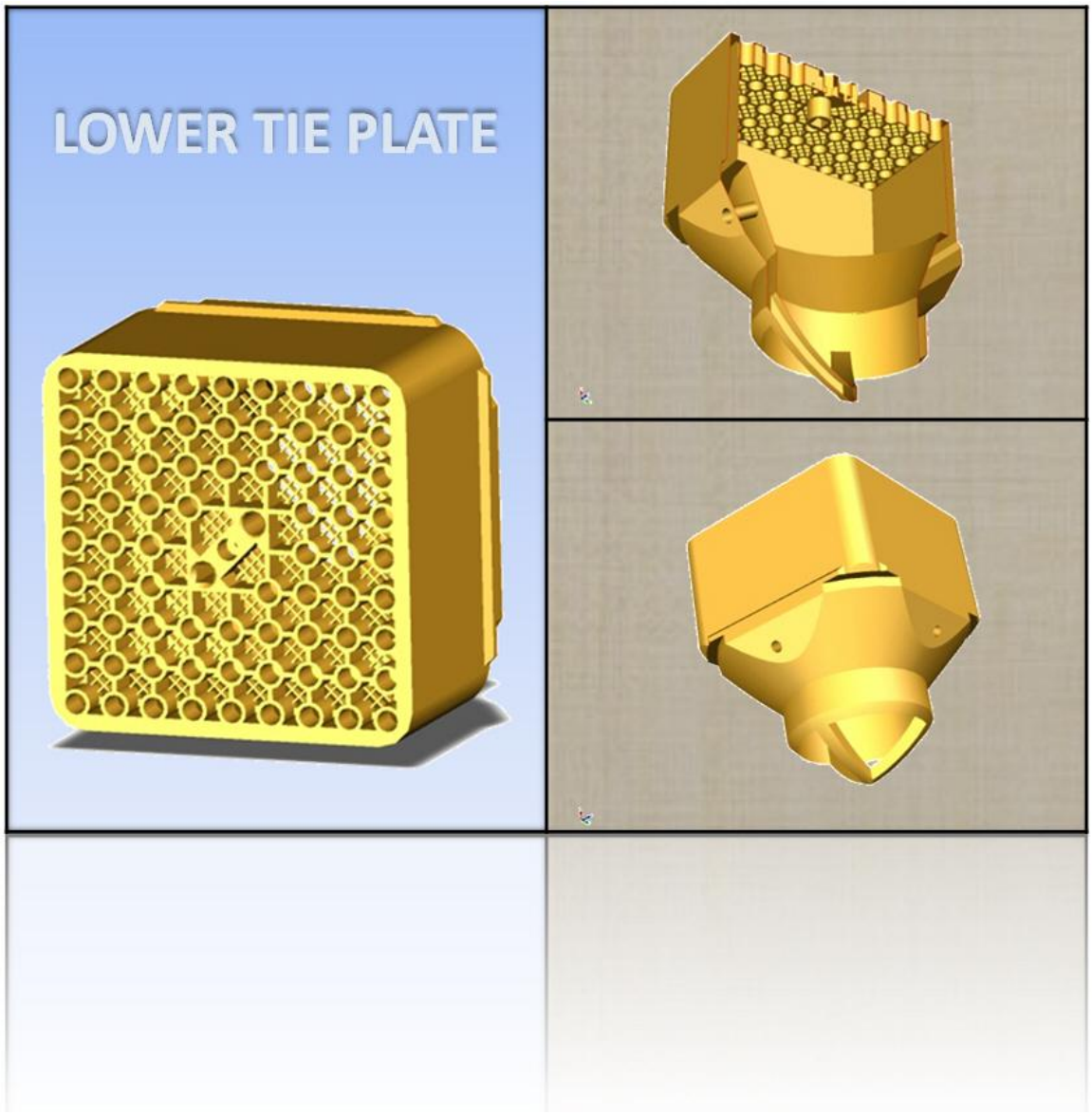


Figure 4.3: CAD solid model of assembly lower tie plate.

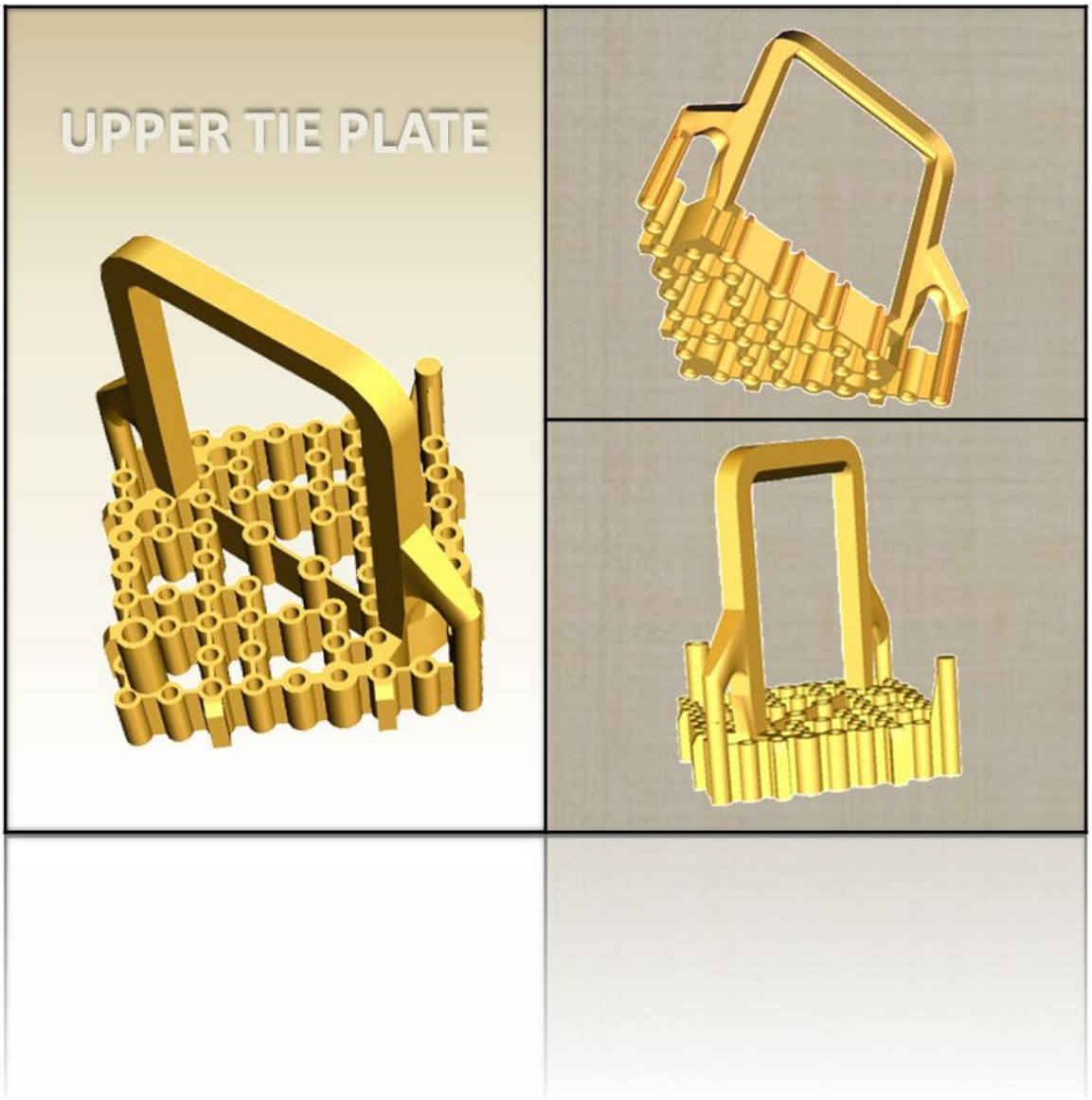


Figure 4.4: CAD solid model of assembly upper tie plate.

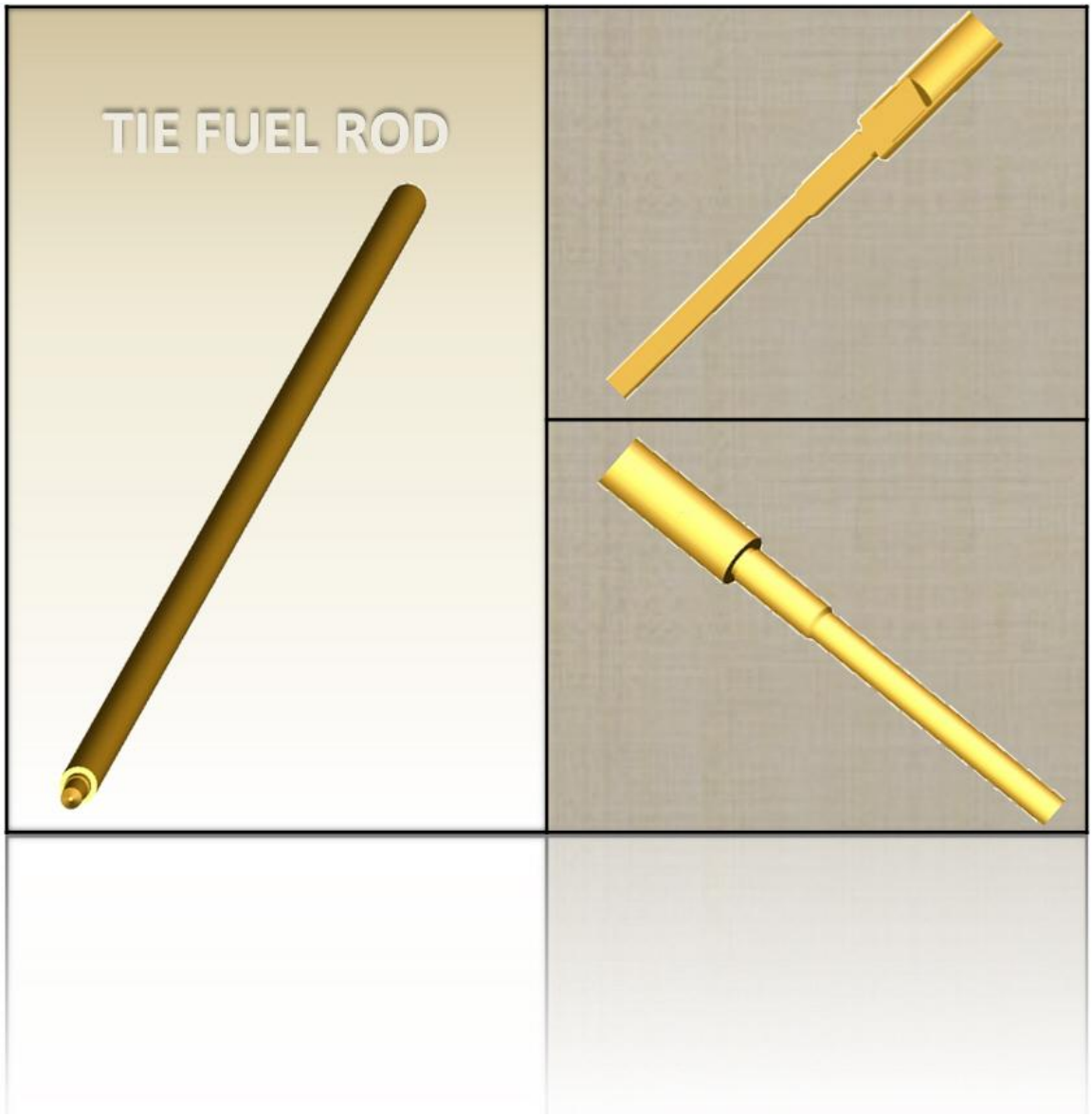


Figure 4.5: CAD solid model of assembly tie fuel rod.

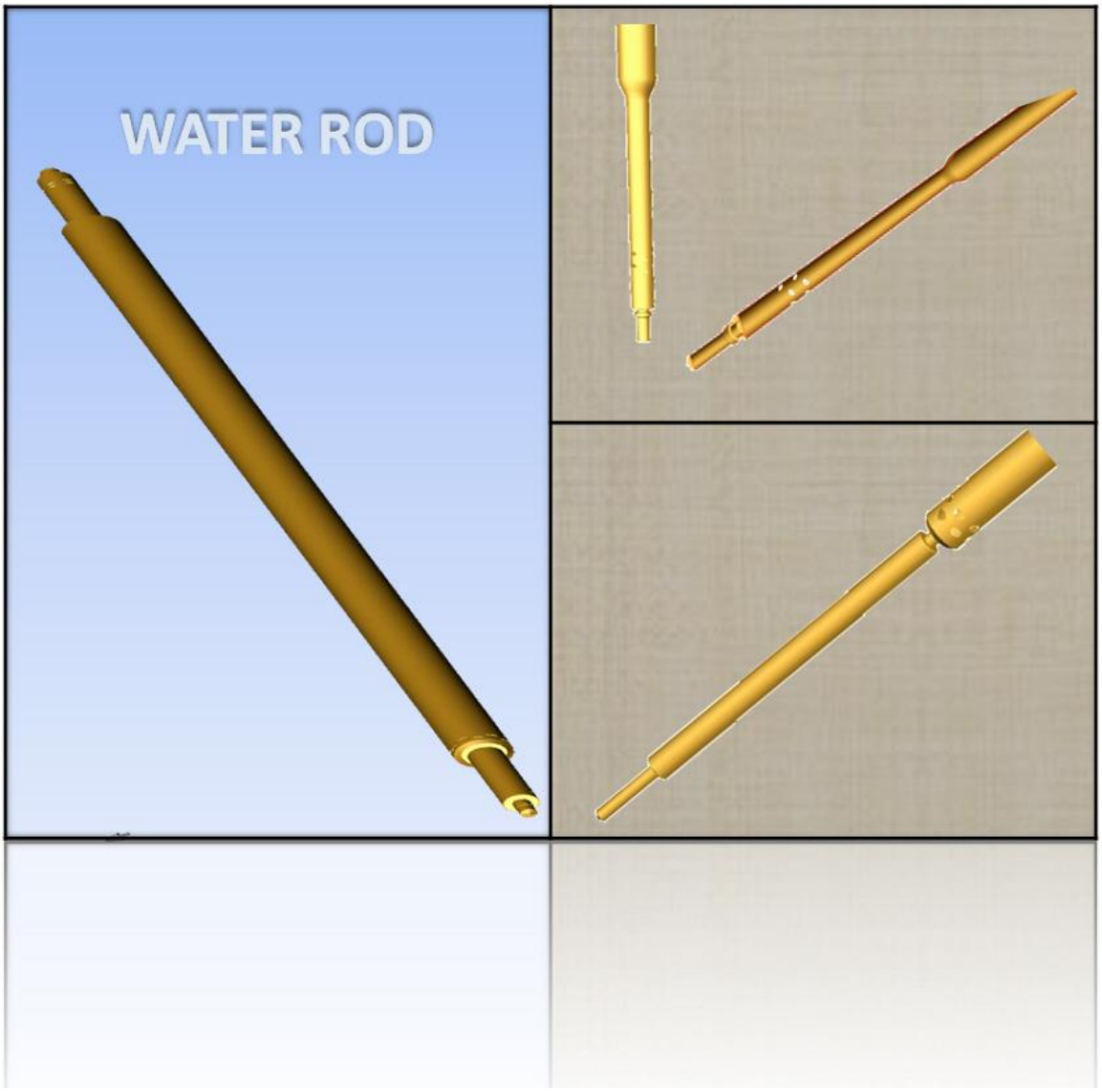


Figure 4.6: CAD solid model of assembly water rod.

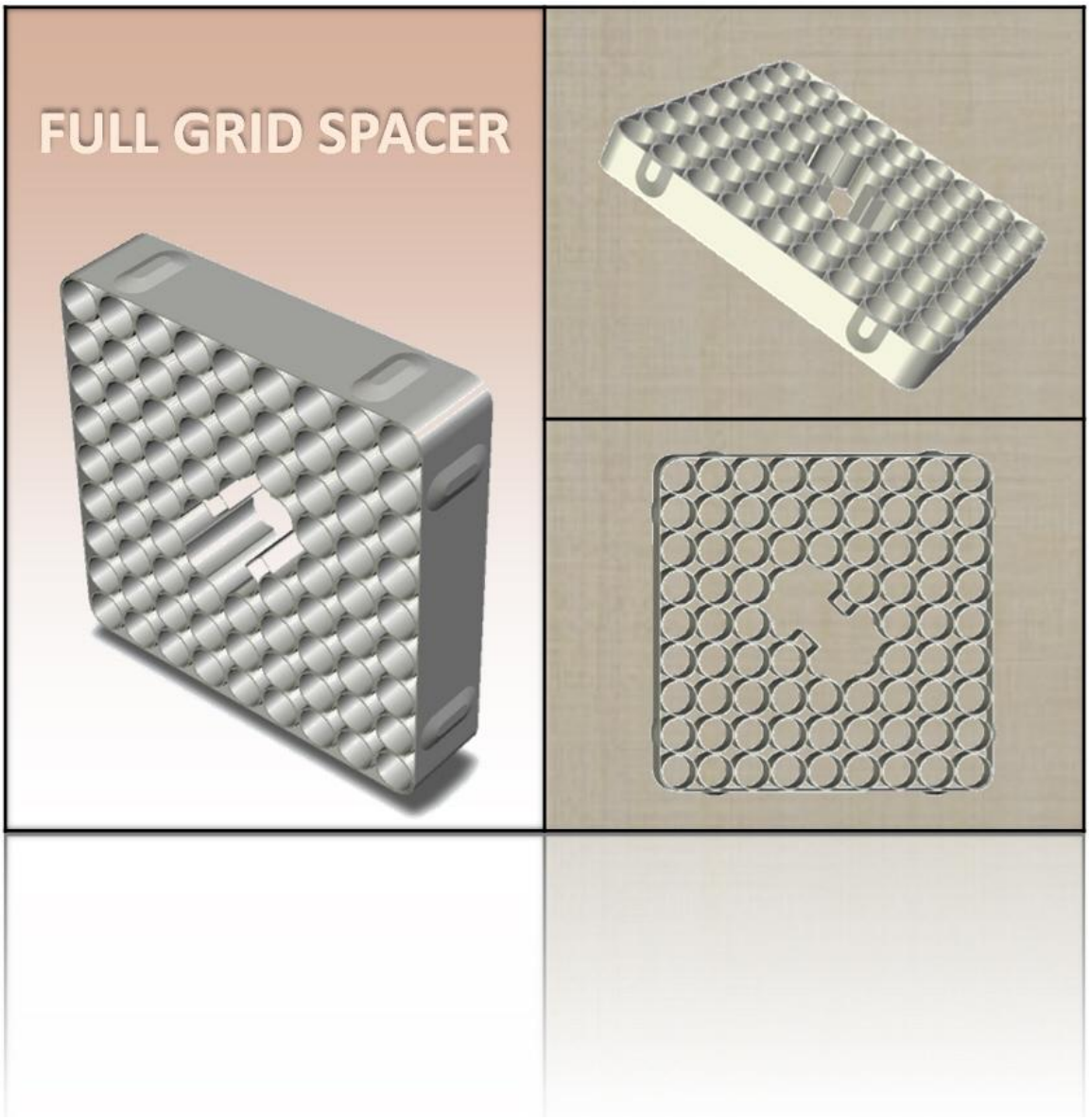


Figure 4.7: CAD solid model of assembly fully-populated spacer.

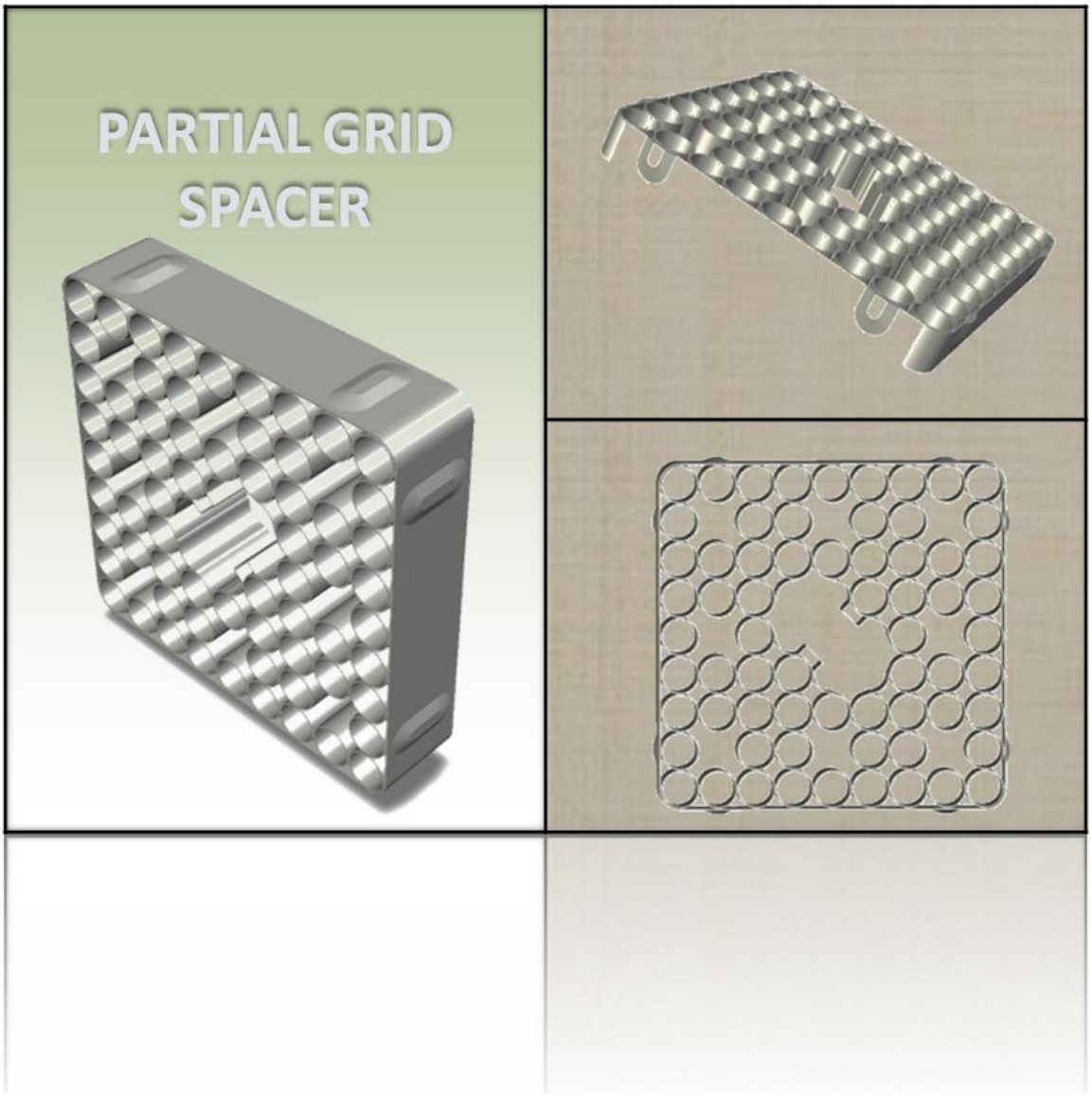


Figure 4.8: CAD solid model of assembly partially-populated spacer.

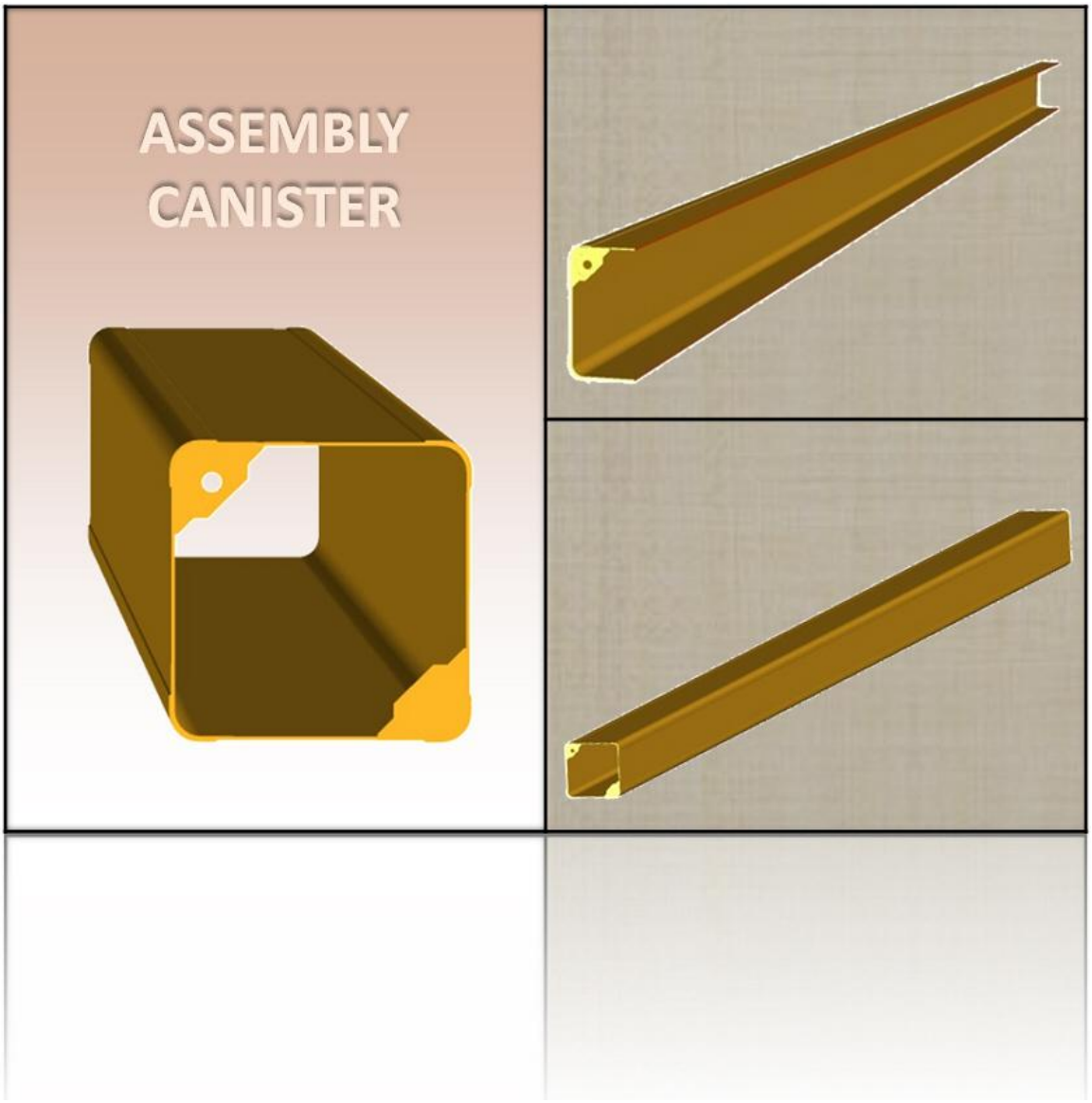


Figure 4.9: CAD solid model of assembly canister.

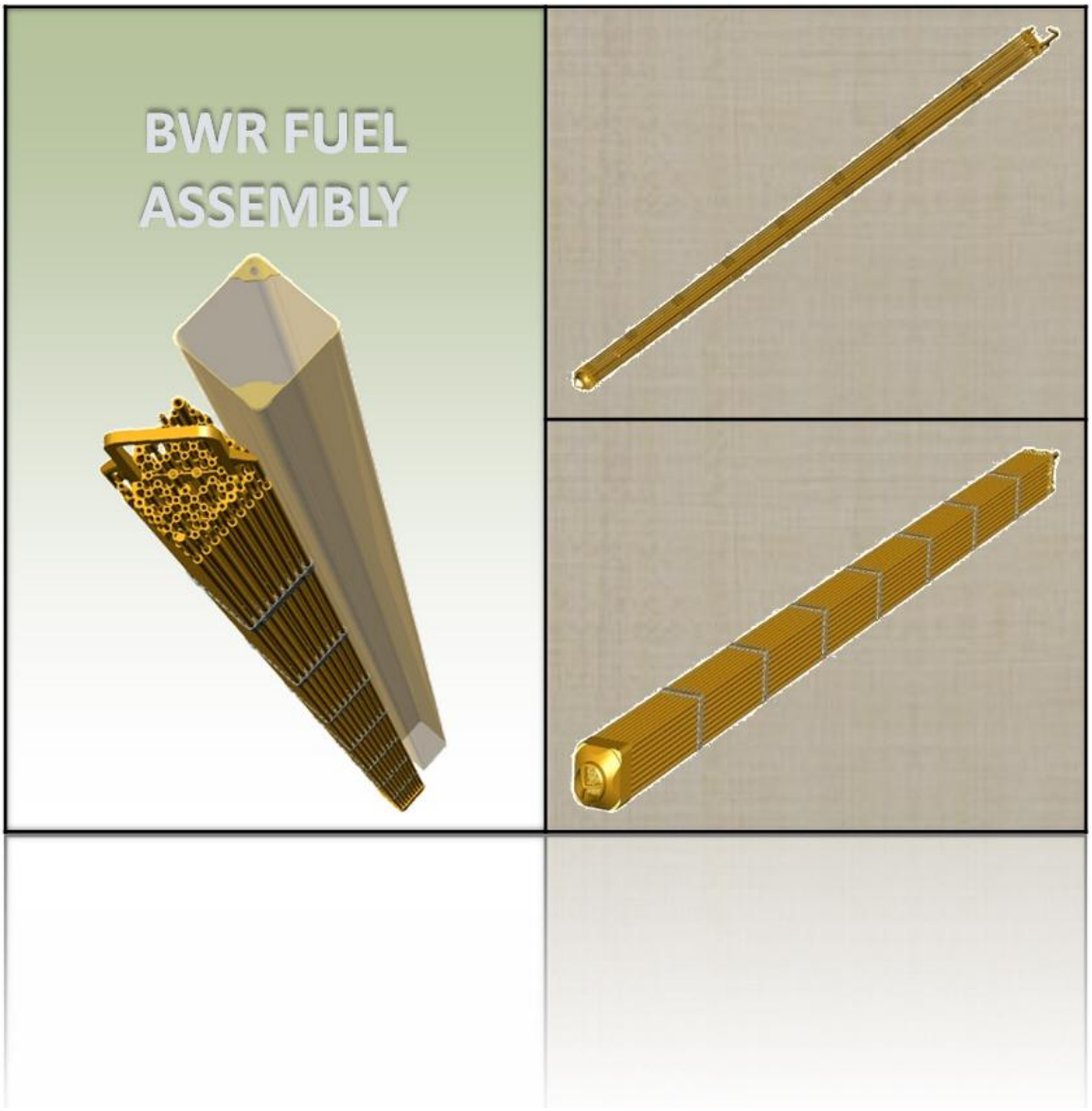


Figure 4.10: CAD solid model of complete BWR fuel assembly.

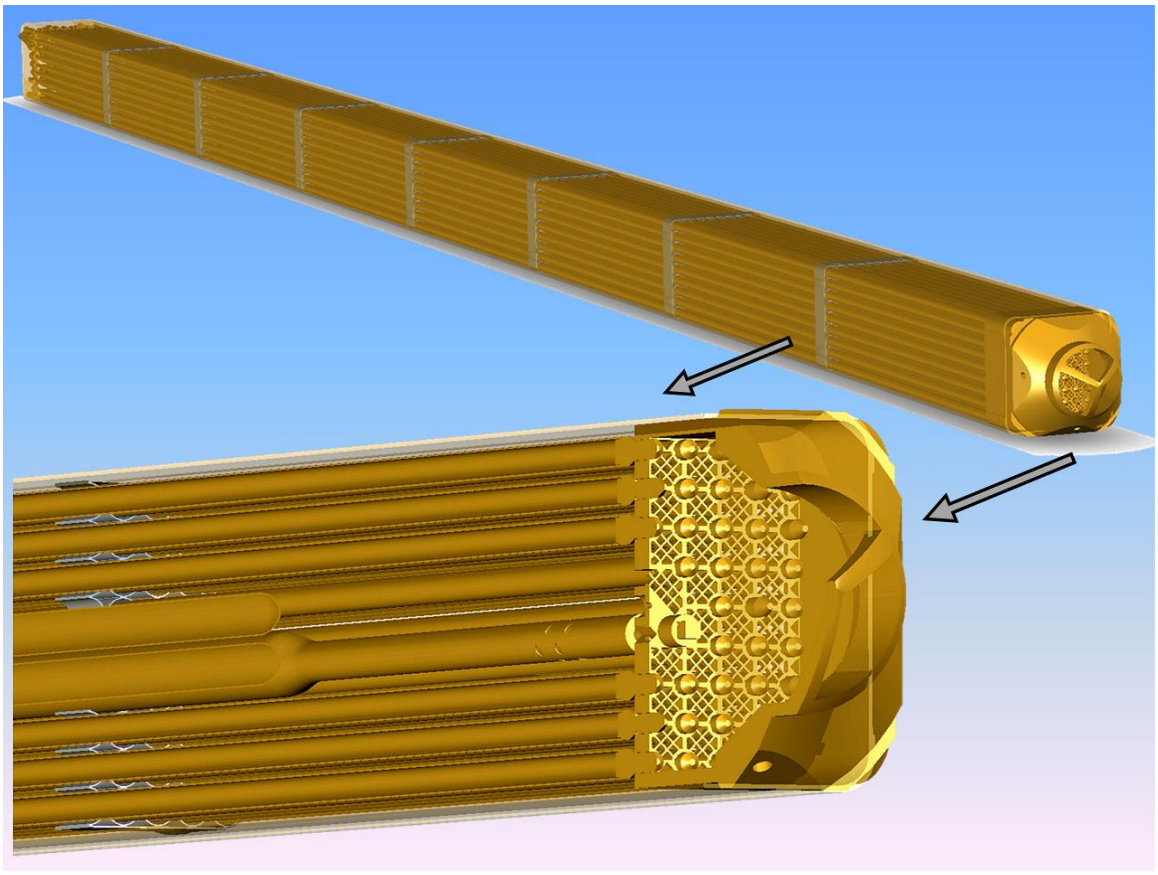


Figure 4.11: CAD solid model isometric and cross-sectioned views of the complete BWR fuel assembly from lower tie plate perspective.

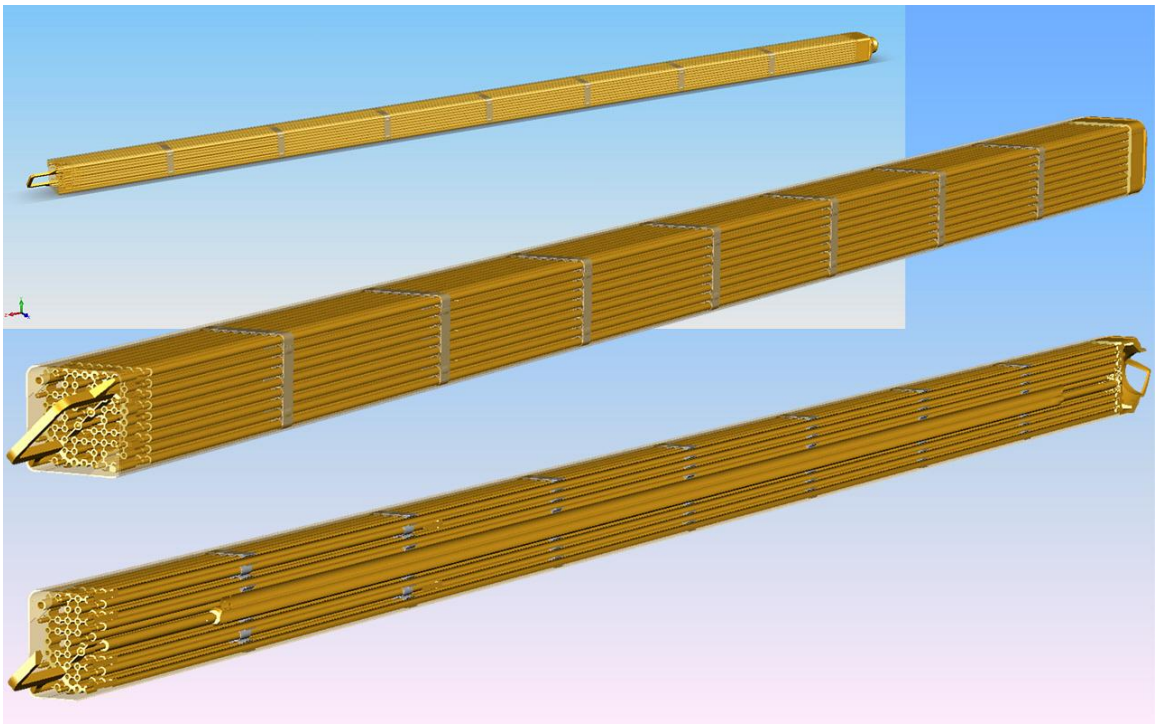


Figure 4.12: CAD solid model isometric and cross-sectioned views of the complete BWR fuel assembly from upper tie plate perspective.

Although each solid model component within the completed assembly CAD model is constructed with an extraordinary amount of detail and precision, a few simplifications are intentionally implemented into the solid model representations of both the fully- and partially-populated assembly spacers. A hypothesis is adopted *a priori*, subject to validation or disproval by the subsequent simulations executed and discussed in Chapter 5, that exclusion of several diminutive-mixing devices (which also function as fuel rod support structures) on these spacers affords a number of substantial benefits to an extent that expected consequential costs are deemed acceptable. The mixing devices expressly omitted from the spacer solid models include the outer grid strip flow deflection vanes (or fingers) and central spacer springs/inner-ring linkages. These

simplifications are illustrated in Figures 4.13 and 4.14 for a fully-populated spacer, but are similarly consistent for the partially-populated spacer as well.

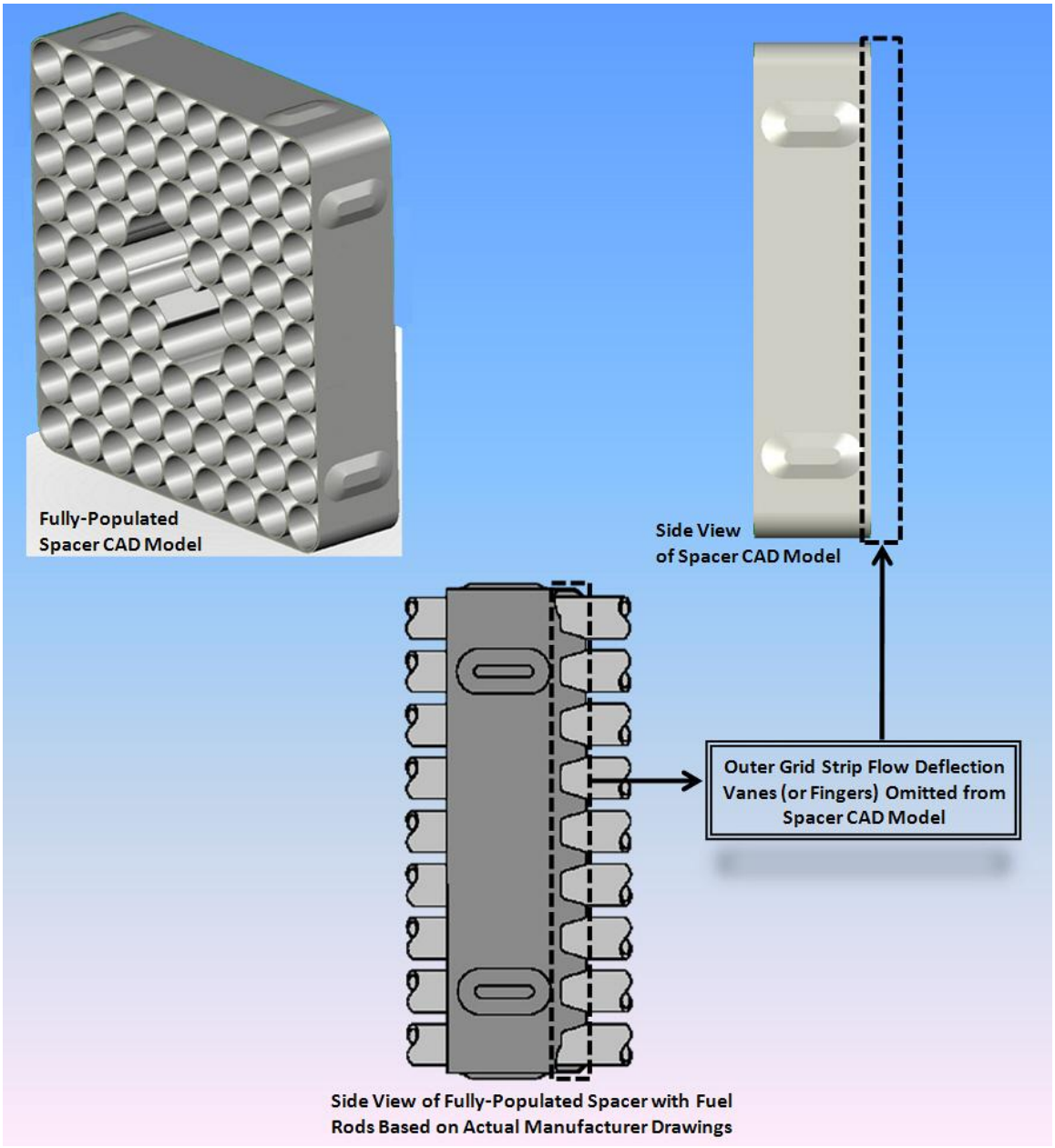


Figure 4.13: Flow deflection vanes on outer grid strip that are omitted from the fully-populated spacer CAD model.

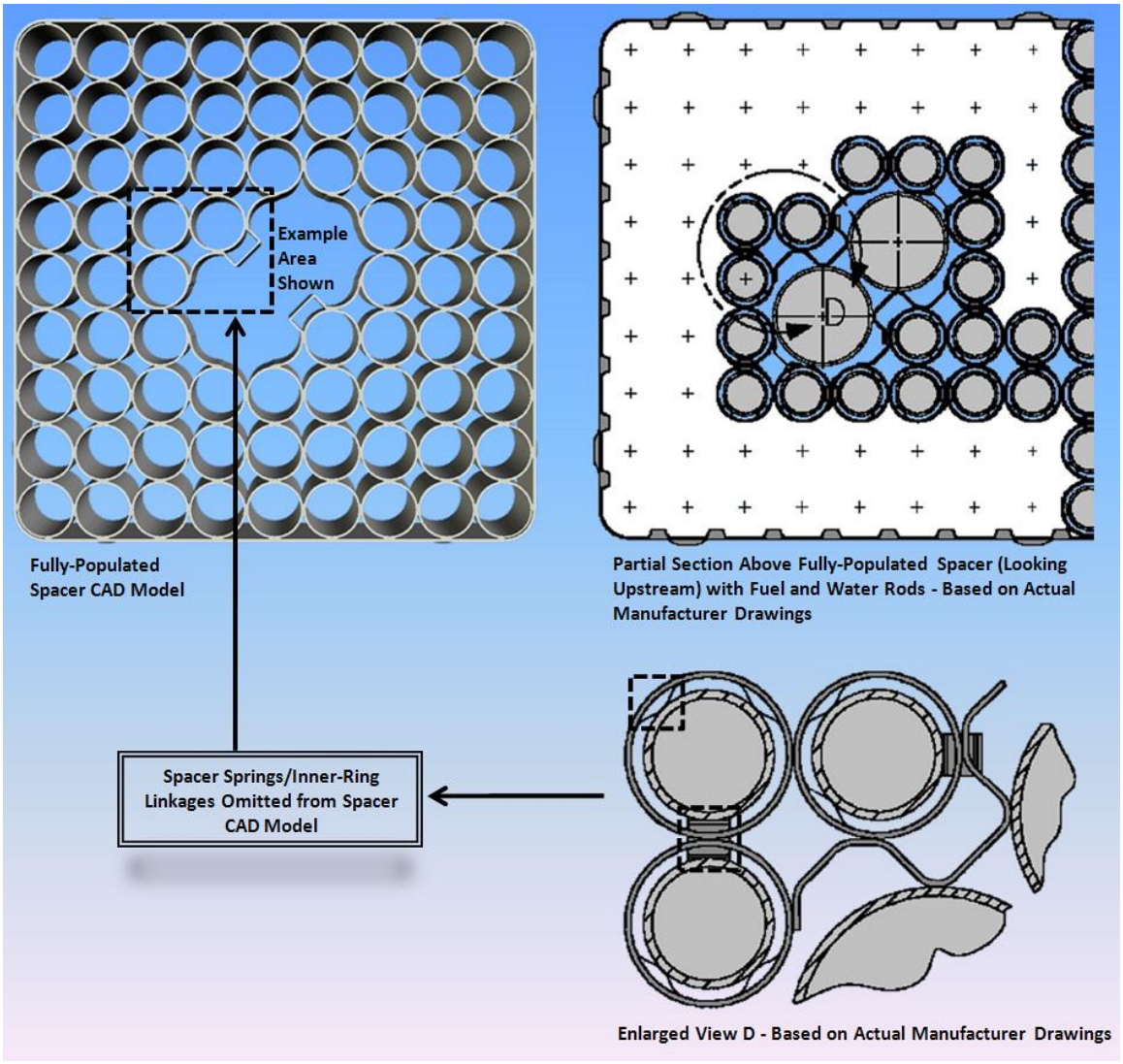


Figure 4.14: Spacer springs/inner-ring linkages that are omitted from the fully-populated spacer CAD model.

The mixing devices omitted in the spacer CAD solid models are characterized by unorthodox 3D surface curvatures that are non-conformal in the axial flow direction with the remaining spacer structure. As a result, meshing the flow volume in the vicinity of these mixing components almost certainly requires the use of unstructured triangular/tetrahedral cell elements, which typically increase total volume cell counts in a

mesh by fifty percent or greater compared with an equivalent quadrilateral/hexahedral mesh for a long, high aspect ratio duct geometry. Numerical diffusion also increases in meshes with triangular/tetrahedral cell elements since the flow is never aligned with the grid and the generally higher skewness values of these elements results in a lower quality mesh as discussed later in Section 4.2.3 of this dissertation. Although difficult, by omitting the mixing devices mentioned from the computational domain and implementing significant decomposition of the volume geometry in the spacer region, the creation of a high-quality mesh composed entirely of quadrilateral/hexahedral cell elements is reasonably achievable. This guarantees alignment of the flow and grid over a substantial portion of the meshed domain utilizing a more moderate number of volume cells as needed for sufficient resolution of flow gradients and minimal error from numerical diffusion.

Despite several physical and computational experiments examining flow through fuel assembly spacers that provide evidence to the contrary (Conner *et al.*, 2010; Ikeda *et al.*, 2006; Toth and Aszodi, 2010), there are key observational differences between these experiments and the accident scenarios investigated in the current research that still suggest the potential validity of this hypothesis. Foremost, the flow conditions in the previous experiments are based on PWR in-core operating conditions in which the coolant fluid medium is water. The Reynolds numbers associated with the evaluated flows are within the approximated in-core range of 70,000 at cold startup to 500,000 at full power, for typical PWR operation (Haslinger *et al.*, 1997). These flow conditions are in stark contrast to the buoyancy-induced airflows expected to develop in SFP complete LOCA scenarios, with Reynolds numbers spanning 70 to 900 within the laminar flow regime (Durbin and Lindgren, 2005). In addition, the conceptual PWR spacer designs incorporated into these research efforts include a larger number of mixing vanes and

spacer springs that are significantly more-pronounced in their impediment of flow as well as ability to incur turbulent mixing downstream of the component, which are quite distinct from current GNF 9x9 BWR assembly spacers in question.

Therefore, considerations for any comparison must include the following: (1) the average viscosity of water is an order of magnitude greater than air over the general operating temperature range; (2) fluid velocities at PWR in-core conditions are between two and ten times higher than those expected for buoyancy-induced airflows in the current research; and (3) flows evaluated in the former experiments are at extensively larger Reynolds numbers. Thus, it is reasonable to assume that amplified spacer effects on turbulence intensity to improve heat transfer characteristics of the fuel assembly along with increased pressure losses observed in previous reference experiments denoting PWR in-core flow conditions are both markedly diminished for BWR spacer designs and buoyancy-induced airflows associated with the current SFP accident research. These deductions provide an adequate basis for rational viability of the previously postulated assumption, which is subsequently evaluated in Chapter 5 with regard to the omission of mixing devices on spacer components in the computational domain model. Nevertheless, the knowledge gained from determining the permissibility of mixing device exclusion on assembly spacer components is an invaluable element for continued research efforts toward a large production-type simulation of an entire fuel assembly (or even group of fuel assemblies).

On a final note, a few additional limitations in the accuracy of the fuel assembly solid model are of noteworthy mention, which are inherent to the uncontrollable circumstances associated with the segregation of computational and physical experiments conducted at The University of Texas at Austin and Sandia National Laboratories, respectively. These limits are encompassed by uncertainties in the variances between

available manufacturer component drawings and actual commercial/mock components used in the physical experiment apparatus as well as several post-computational model development modifications integrated into the final SNL experimental setup. Specifically, these uncertainties arise primarily because individual components used to construct the SNL prototypic fuel assembly are to some extent a compilation of similar but faintly amended parts from different GNF/GE11 9x9 BWR fuel assembly design revisions, in which distinguishing specifications of the varying components are not disclosed by the manufacturer. Furthermore, the dimensioned drawings acquired by SNL do not necessarily reflect all of the purchased commercial components made-available by GNF and included the understanding that one or more parts are sold based on the premise of possessing a slight defection. This in essence offers some protection of proprietary information, which is one of the manufacturer's biggest concerns in allowing SNL to purchase the commercial fuel assembly components (personal communication with Eric Lindgren at SNL, June 2005).

Unfortunately, the proposed timeline for conducting computational experiments required completion of the fuel assembly solid model over three months prior to the scheduled assembly of the final experimental apparatus, which precluded any possibility of resolving potential variances between final assembly components not yet obtained from the manufacturer and the related dimensioned drawings provided by SNL. Although a substantial effort is expended toward the alleviation of any modeling inconsistencies, discrepancies inevitably exist between the CAD solid model of the fuel assembly and the actual prototypic experimental assembly at SNL. However, it is assumed that geometrical differences are largely negligible and further attempts to quantify the effects on resulting simulation trials are not immediately considered in the current breadth of research.

4.2 COMPUTATIONAL FLOW DOMAIN FROM CAD SOLID MODEL

CFD numerical experiments conducted in this research include a multitude of simulation trials that employ one of three computational mesh models, each possessing different domain geometry and extent of the fuel assembly. The following discussions serve as a precursor to the elucidation of specific details in Chapter 5 with regard to the domain geometry, mesh generation, FLUENT solver setup, and flow conditions associated with each of these models. Particular emphasis is focused upon the reasoning for selecting the fuel assembly segments modeled, use of CAD/CAE filters in GAMBIT for importing solid surface geometries of fuel assembly components from the CAD solid model, and the general aspects of mesh generation including requirements, topology, and quality. However, only the two computational flow domain models explicitly constructed from the actual CAD solid model geometry of the fuel assembly, which include the single fully-populated fuel bundle run and fuel bundle run with spacer models, are expressly considered.

The remaining full-length equivalent model of a fuel assembly is not based on (or constructed from) any segment of the actual physical geometry represented by the CAD solid model. It is simply an equivalent representation of the hydraulic characteristic values and length scales pertaining to the actual fuel assembly geometry, but incorporates an extremely modest shape for the cross-sectional area of flow in lieu of genuine assembly component structures. The simplicity of the model geometry permits direct construction of the flow domain within the GAMBIT preprocessor software without the use of a CAD solid model. A substantially different design methodology combined with simplicity of the computational domain construction warrants appropriate, yet brief

description outside the subject matter of this chapter; therefore, adequate details of the mesh model are discussed along with the CFD simulation trials implementing this particular flow domain in Section 5.3. On the other hand, the general aspects of mesh generation in Section 4.2.3 below are applicable to all three computational flow domain models.

4.2.1 Fuel Assembly Segments Modeled

The first two phases of numerical experiments conducted in this research are congruent with the initial stages of progression for simulating and validating increasingly larger-scale computational flow domain models of the fuel assembly. As previously discussed in Sections 1.4 and 1.5 above, the absence of any former, supporting attempts to model BWR assembly geometry at a high fidelity under the postulated accident conditions necessitates an increased effort in order to assure the validity, application, and accuracy of results with respect to the FLUENT CFD models employed. Specifically, no prior numerical modeling data exists to support any assumptions of noteworthy significance in the computational model development, and the research evaluation process must start from a relatively primitive beginning.

Therefore, a substantial effort is expended toward the assessment of basic constraints associated with the development of model geometry, mesh generation, solver setup, and computational requirements beginning with a phase of preliminary scoping studies over smaller segments of the fuel assembly model. Again, this is an integral practice in the appropriate progression of simulating and validating large production-type problems in the initial model development phase in order to identify numerical uncertainties and potential challenges on smaller mesh sizes prior to the incorporation of

prohibitively larger-scale mesh models in which the resolution of modeling issues are significantly more problematic. The assembly flow segment modeled in the first phase of numerical experiments conducted includes the most prevalent cross-sectional geometry across the greatest extent of the fuel assembly length. This model, termed the single fully-populated fuel bundle run, is comprised of a centered, two-thirds length equivalent segment of a fully-populated fuel bundle run between two fully-populated spacers. An example segment location within the fuel assembly and enlarged cross-sectional view of this modeled domain are shown in Figure 4.15 below.

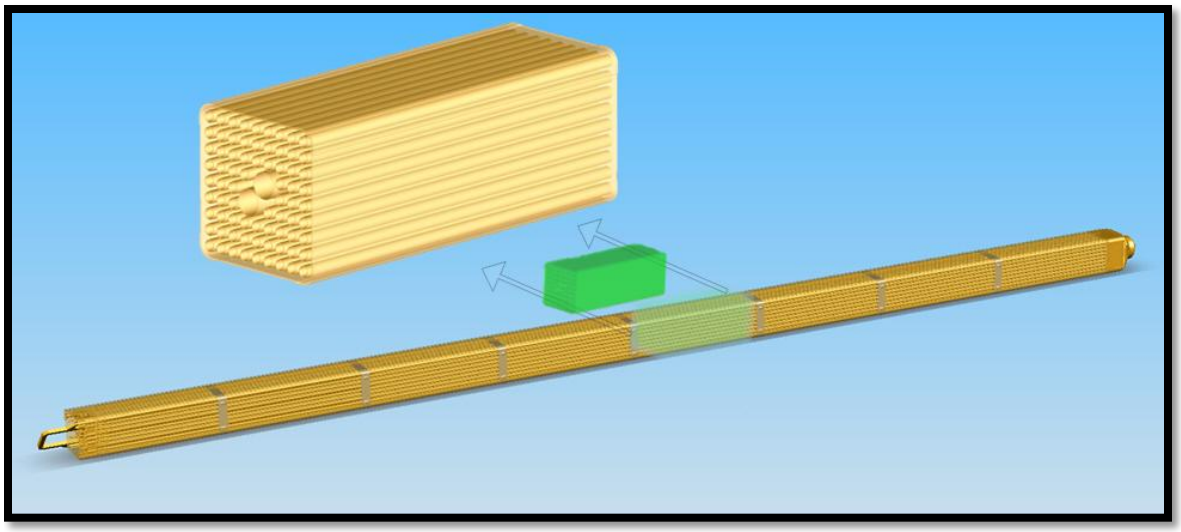


Figure 4.15: Example fuel assembly segment representing the single fully-populated fuel bundle run model used in the first phase of numerical experiments.

Several of the preliminary scoping trials of specific importance include the gauging of computational resource requirements, grid-independence, periodicity and symmetry applicability, validity of viscous model options and wall treatment method (if necessary), appropriateness of boundary conditions, and other pertinent CFD model characteristics which aid in establishing sufficient accuracy and confidence in current as

well as subsequent model developments employed. A substantial number of these validation and parametric comparisons are reasonably achievable with moderate computational resources by reducing the size of the fuel assembly segment considered in these beginning case studies. Due to the sizeable proportion of the entire fuel assembly length embodied by fully-populated fuel bundles of corresponding segment geometry, the use of this flow domain selection is deemed most appropriate for preliminary modeling analyses.

Knowledge gained from the initial trial simulations and comparison of results with hydraulic analysis data from SNL are incorporated into the subsequent development of a larger flow domain model utilized in the second phase of numerical experiments conducted. This model, termed the single fully-populated fuel bundle run and spacer, encompasses an entire fully-populated fuel bundle run and spacer segment of single-span length from the prototypic experimental fuel assembly. The division of the computational flow domain occurs at the midpoints of consecutive fully-populated bundle runs, each spanned by fully-populated spacer components at the upstream and downstream boundaries. An example extraction of the described flow segment from the fuel assembly apparatus including an enlarged cross-sectional view of this modeled domain are illustrated in Figure 4.16 below.

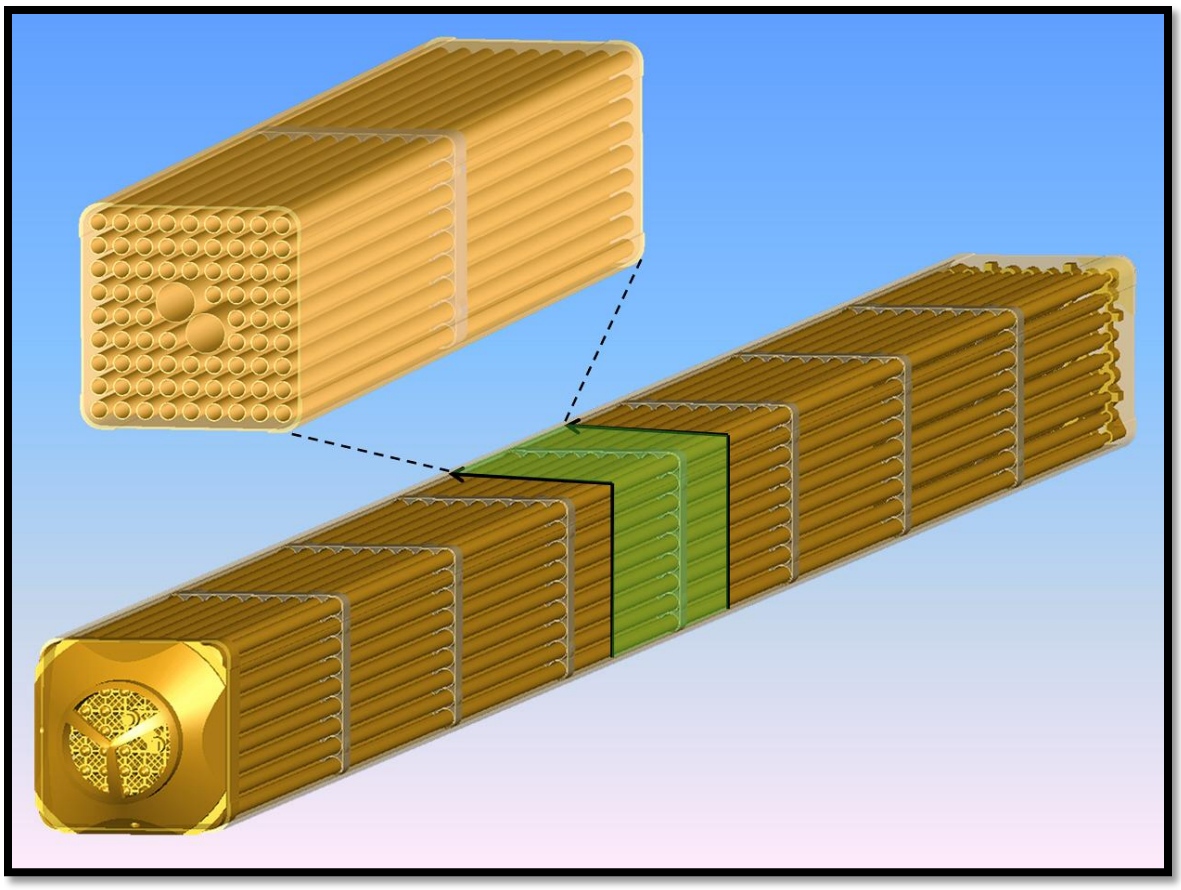


Figure 4.16: Example fuel assembly segment representing the single fully-populated fuel bundle run and spacer model used in the second phase of numerical experiments.

The second phase of simulation experiments employing the extended flow domain section of the fuel assembly shown in Figure 4.16 is simply part of the progression towards eventually modeling the flow domain of the entire 9x9 BWR fuel assembly. Periodic repetition of the fuel assembly geometry captured by this model embodies the greater portion of the complete cross-sectional area of flow traversing the axial length of the actual fuel assembly and is obviously a justifiable selection for incrementing the size of the flow domain modeled. Again, knowledge gained from this second phase of research and the comparison of results with hydraulic analysis data from SNL are

intended to direct and aid future research modeling efforts beyond the objectives accomplished at the stage of research presented in this dissertation. Although more-specific details with regard to the geometry and hydraulic flow characteristics of each computational flow domain model are provided relative to their discussion in Chapter 5, a few general dimensions and hydraulic characteristics of the two described models are listed in Table 4.1 below for reference.

Table 4.1: General dimensions and hydraulic characteristics of both the first and second phase research CFD models.

FIRST AND SECOND PHASE RESEARCH CFD MODEL CHARACTERISTICS	
Single Fully-Populated Fuel Bundle Run Model	Value:
Flow Area, (m^2)	0.00968829
Hydraulic Diameter, (m)	0.01190721
Total Model Length, (m)	0.32088667
Single Fully-Populated Fuel Bundle Run and Spacer Model	Value:
Flow Area [Fuel Bundle], (m^2)	0.00968829
Flow Area [Spacer Midplane], (m^2)	0.00726749
Hydraulic Diameter [Fuel Bundle], (m)	0.01190721
Hydraulic Diameter [Spacer Midplane], (m)	0.00290797
Fuel Bundle Length, (m)	0.481330
Spacer Length, (m)	0.030480
Total Model Length, (m)	0.511810

4.2.2 Solid to Flow Domain Model Transition

Upon completing the CAD solid model of the experimental fuel assembly at SNL and segmenting the flow domain desired for modeling from this larger assembly model, a suitable file format for saving the CAD geometry is required to implement the built-in CAD/CAE filters in GAMBIT for importing the solid surface geometries of the

associated flow domain components. Identifying accurate and efficient transferable file formats between the CAD modeling and grid-generating software that allow direct import of the solid surfaces to form a real Boolean operable volume(s) is critical for establishing an acceptable flow domain for meshing. Several of the most popular file formats that are widely known, highly interchangeable, and supported by both software packages include ACIS (*.sat), IGES (*.igs), STEP (*.step), and Parasolid (*.x_t or *.x_b).

A number of trial and error attempts to import, cleanup, and utilize each of the above file types with various model components comprising the BWR fuel assembly concluded the ACIS version 8.0 and Parasolid file formats as the most appropriate for exporting and importing solid surface geometry from SolidWorks into GAMBIT, respectively. These file formats afford the user minimal geometry cleanup (if any) and a quick turnaround time for generating the flow volume for ensuing mesh operations prior to the exportation of the final computational grid to the FLUENT flow solver. The actual export/import process is rather straightforward and trivial, provided the CAD model design is genuinely of good quality and no extenuating errors or inconsistencies exist.

In such cases, opting for scripted healing and tolerance correction greatly expedite flow domain construction, permitting the software to automatically attempt to form one or more ACIS or Parasolid bodies even if minute discrepancies in the form of small cracks, duplicated geometry, or incomplete surface representation by mathematical function(s) are determined to subsist within the CAD model file. In other words, these options aid in the construction of volume bodies with 'water-tight seals' by amending geometrical entities perceived to contain boundary errors within a certain user-set percentage (FLUENT Inc., 2006). A few example images of solid model components following import into GAMBIT, utilizing the ACIS and Parasolid file formats from import tests conducted, are provided in Figures 4.17, 4.18, and 4.19 below.

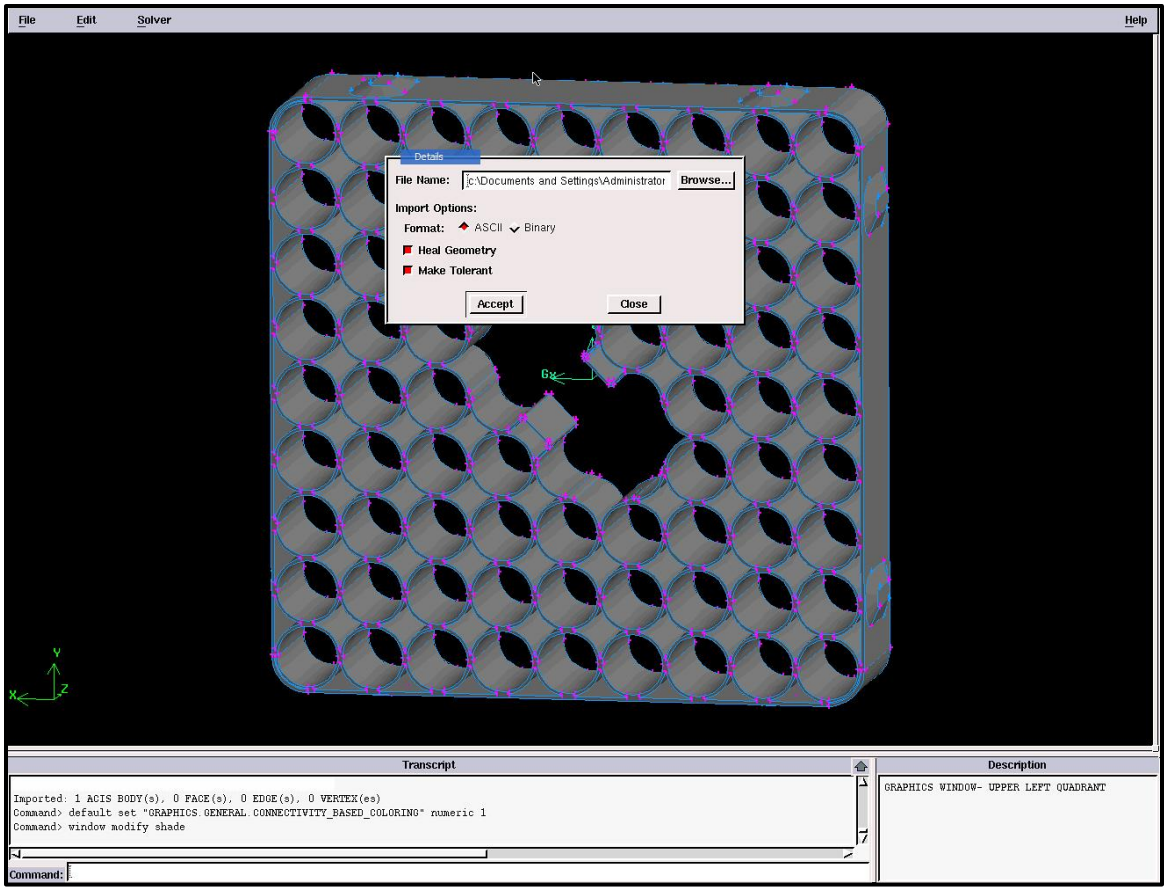


Figure 4.17: Snapshot of fully-populated spacer component imported into GAMBIT from SolidWorks CAD solid model.

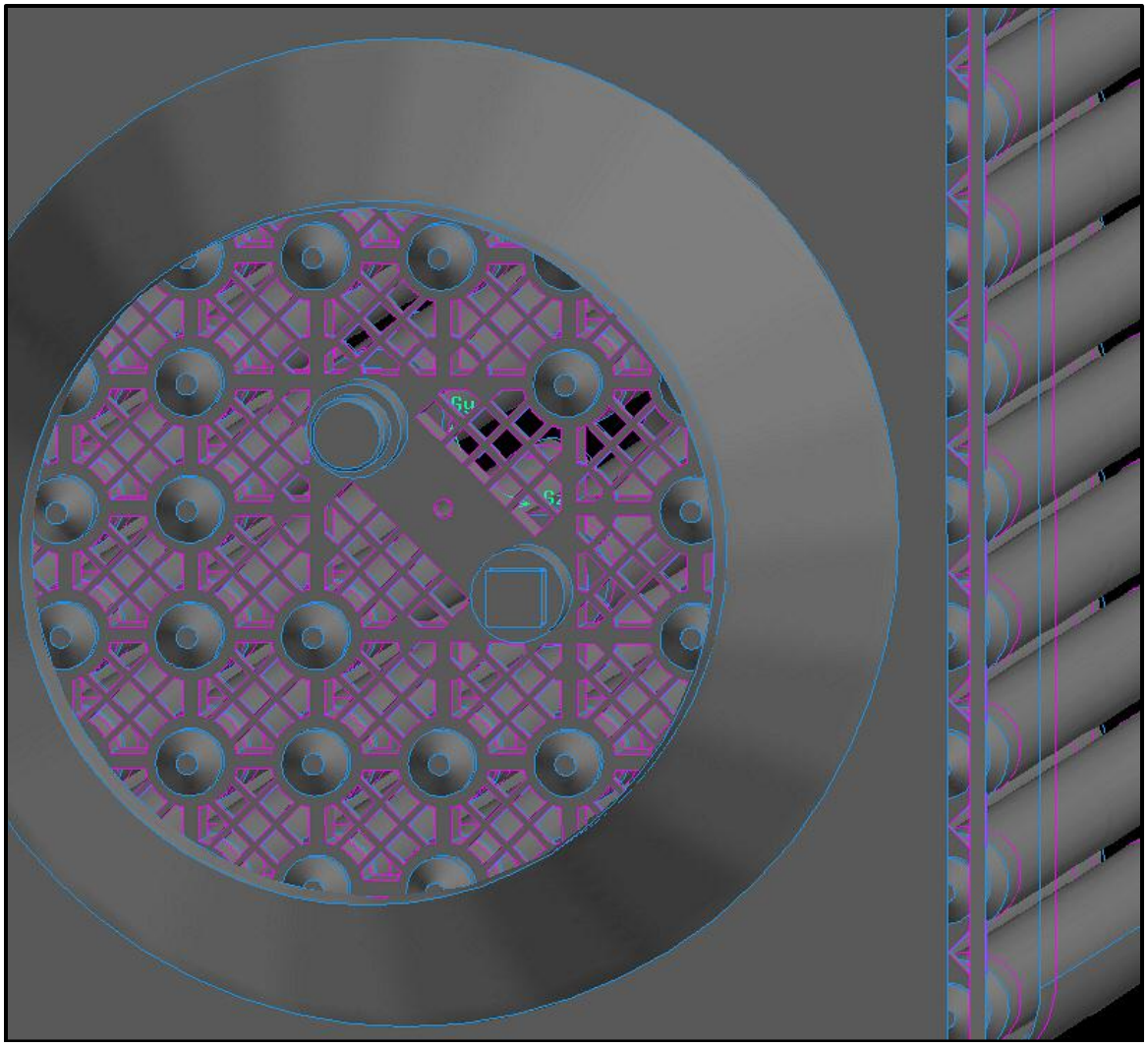


Figure 4.18: Lower fuel assembly segment including lower tie plate and fuel rods imported into GAMBIT from SolidWorks CAD solid model; several surfaces are made transparent in order to enhance the visual detail.

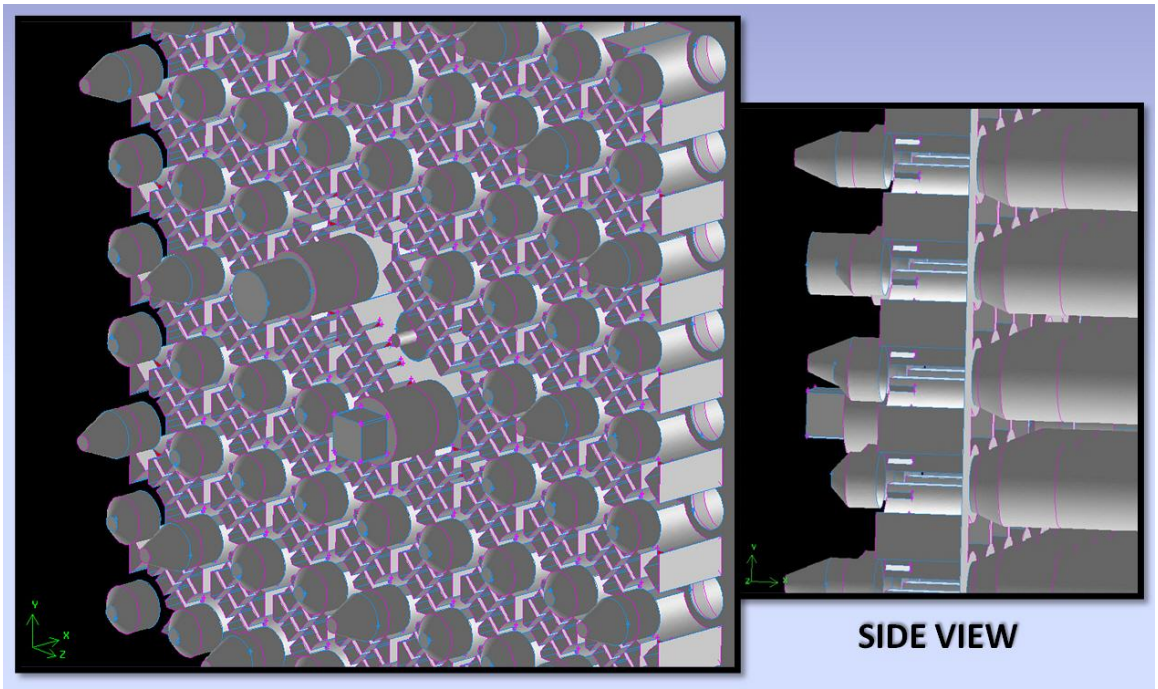


Figure 4.18: Isometric and side views of rod connections at the debris screen of the lower tie plate from a lower fuel assembly segment imported into GAMBIT from SolidWorks CAD solid model; several surfaces are made transparent in order to enhance the visual detail.

After the solid surface geometry associated with the fuel assembly components modeled within the flow domain are imported and checked for validity in GAMBIT, the actual flow volume is constructed from these bounding surfaces using built-in geometry tools and Boolean operations within the GAMBIT preprocessor software. Thus, the next step in fabricating both the single fully-populated fuel bundle run and fuel bundle run with spacer models requires the development of a bounding volume that encompasses the entire domain modeled. This is accomplished by stitching the fuel assembly inner canister surfaces and normal flow end faces together to form a single volume.

Subsequently, the final flow domain is formed by Boolean subtraction of all internal solid component subvolumes (*i.e.* fuel rods, water rods, and spacer solid bodies

created upon importation) from the stitched bounding volume that results in a new equivalent volume representing only the interstitial flow space within the modeled fuel assembly segment. Noting the geometric and presumed fluid flow symmetry of the computational domain in both research phase models, the flow volume is further divisible into four equal subsections defined by two diagonally-edged planes traversing the volume segment from the inner canister surface corners through center point of the fuel assembly. As a visual reference, the computational flow domain with dividing symmetry planes of the single fully-populated fuel bundle run model is shown in Figure 4.19 below.

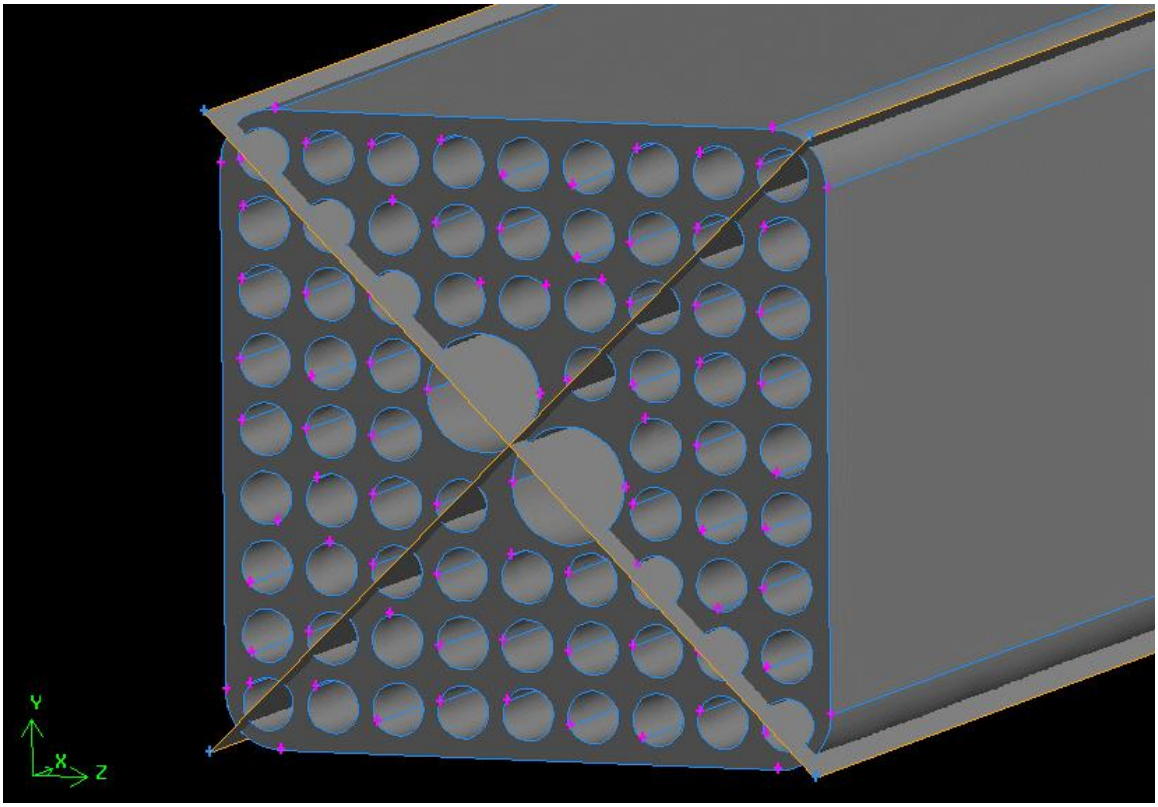


Figure 4.19: Computational flow domain with dividing symmetry planes of the single fully-populated fuel bundle run model.

In conclusion, any three of the four subvolumes formed by the dividing planes (as shown in Figure 4.19) are deleted in both research phase models to complete the final flow volume desired for mesh generation and implementation into FLUENT CFD simulation trials (excluding the symmetry validation study discussed in Section 5.1). The inner fuel assembly faces at flow volume boundaries in each model, which are created by the dividing symmetry planes, are subsequently designated as symmetry boundaries. This is illustrated in Figure 4.20 below for the flow domain model of the single fully-populated fuel bundle run. In addition, the computational flow domain of the single fully-populated fuel bundle run and spacer model (prior to the removal of any symmetry subvolumes) is presented in Figure 4.21 along with decomposition of the flow domain into several subsegment flow volumes surrounding the grid spacer component for enhanced detail.

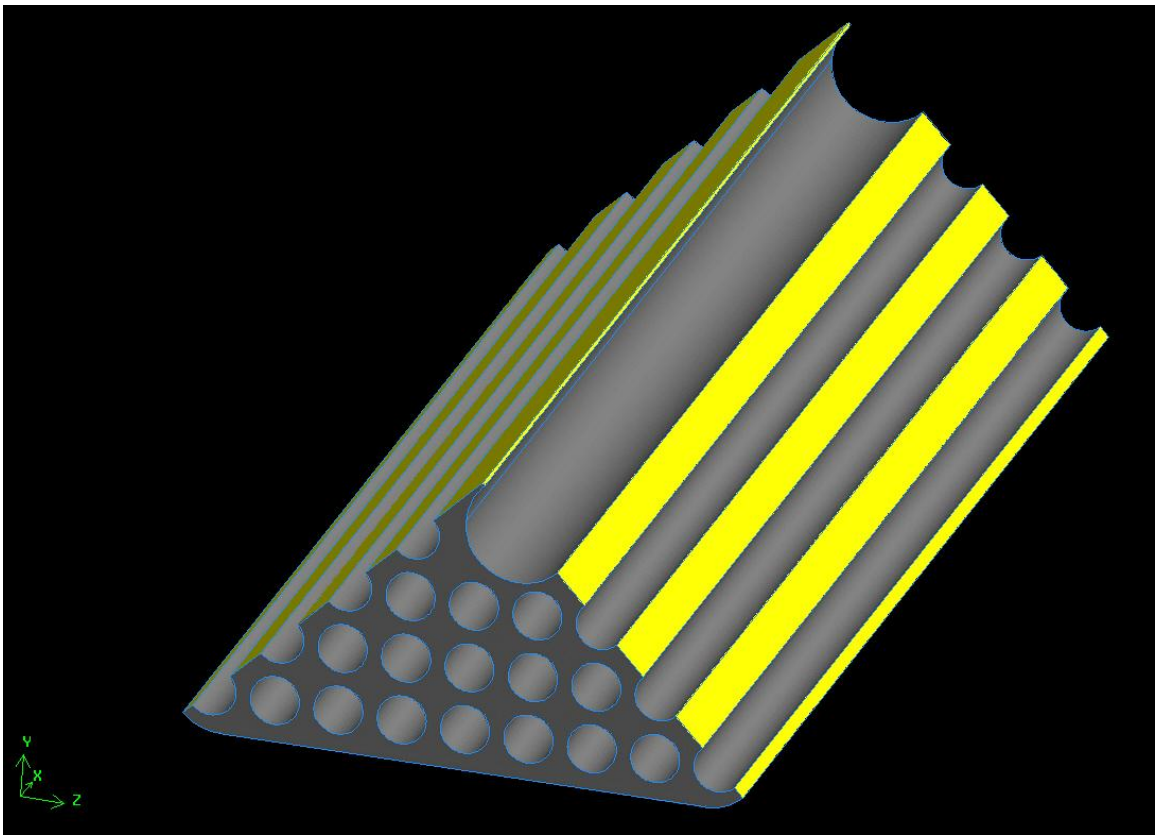


Figure 4.20: Flow domain model of the single fully-populated fuel bundle run with symmetry subvolumes removed and symmetry boundaries highlighted in yellow.

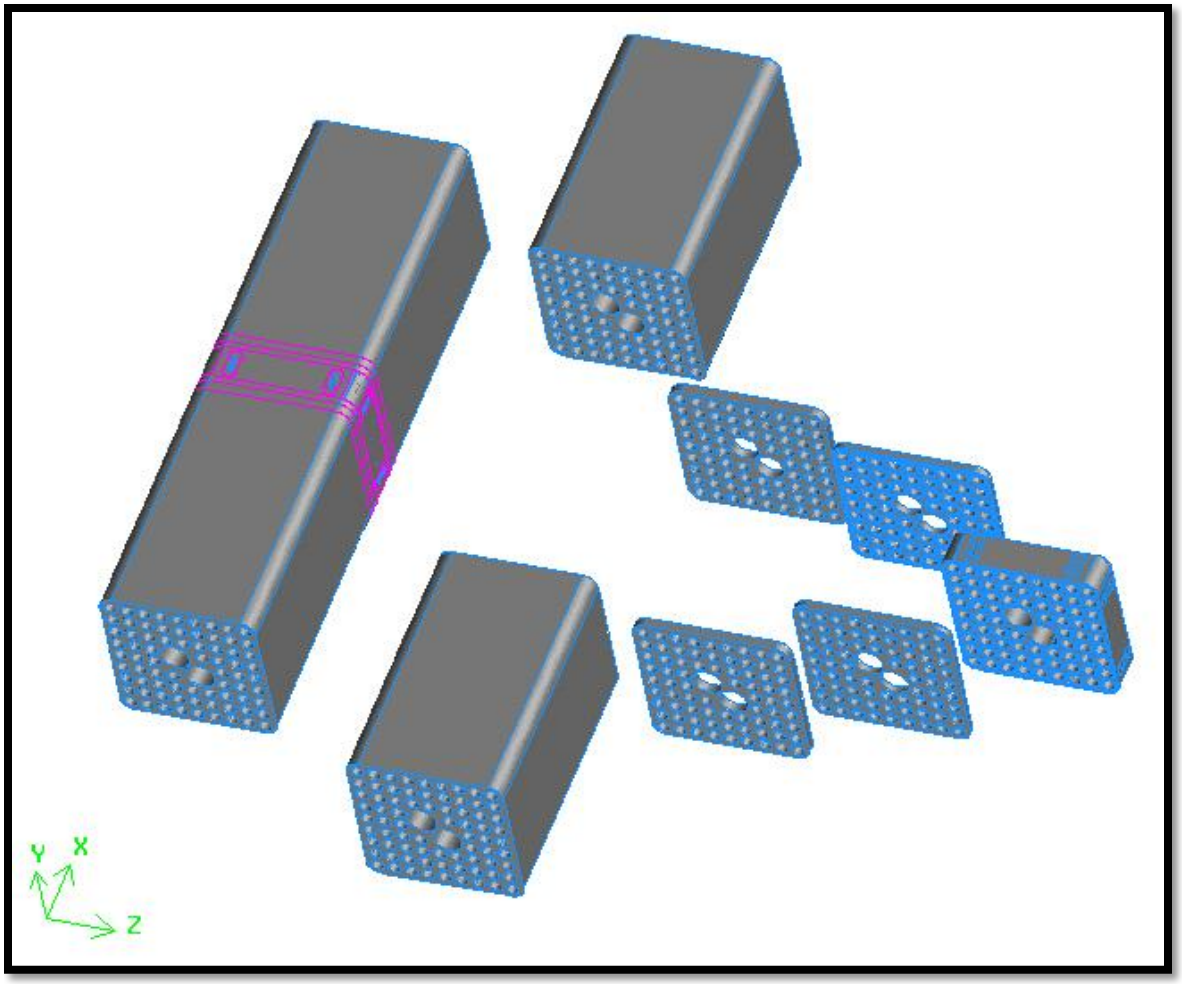


Figure 4.21: Computational flow domain of the single fully-populated fuel bundle run and spacer model (without application of symmetry boundaries) along with moderate decomposition of the flow domain for enhanced detail.

4.2.3 General Aspects of Mesh Generation

The generation and properties of the mesh characterizing the computational flow domain used in each numerical simulation are specifically considered in accordance with the type of viscous model selected in the FLUENT flow solver setup. This is due to the near-wall treatment and mesh requirements at bounding walls which "significantly

impact the fidelity of numerical solutions, inasmuch as walls are the main source of mean vorticity and turbulence" (FLUENT Inc., 2006) as well as viscous pressure loss in both laminar and turbulent flows. A number of considerations concerning the near-wall treatment options and mesh requirements associated with turbulence models in the FLUENT CFD software are previously addressed in Chapter 2 of this dissertation. Although an exact near-wall mesh requirement does not similarly exist for the laminar viscous model, the FLUENT user's guide suggests the following placement of wall-adjacent cells in laminar flows (FLUENT Inc., 2006):

$$y_p \sqrt{\frac{u_\infty}{\upsilon x}} \leq 1 \tag{4.1}$$

where, (y_p) is the distance between wall and adjacent cell centroid, (u_∞) is the free-stream velocity, (υ) is the kinematic viscosity of the fluid, and (x) is the distance along the wall from the starting point of the boundary layer.

As with the geometric and hydraulic flow particulars, the specific details with regard to the properties and characteristic values associated with the computational grid(s) generated for each individual flow domain modeled are provided and discussed at length in Chapter 5 sections corresponding to the relating simulation trials. The primary intent of the following discussion is simply to introduce general aspects of mesh generation considered/applied for creating the grid(s) employed in representing the computational flow domains modeled in this research. In particular, focus is directed upon issues surrounding the subjects of grid topology selection and mesh quality influence on CFD computations.

Identifying the most appropriate grid topology (or mesh type) is largely application specific and requires deliberation of research constraints associated with an acceptable amount of setup time, computational resource expense, and numerical diffusion introduced into the resulting solution. In general, unstructured grids using triangular/tetrahedral elements provide a quick and easy mesh solution for flow domains involving complex geometries where structured or block-structured grids consisting of quadrilateral/hexahedral elements are difficult to construct and very time consuming to employ. However, for a long narrow-ducted volume or equivalent form of geometry (such as the modeled domains of interest in this research) it is preferable to generate a quadrilateral/hexahedral element mesh that aligns well with the flow and shape of the domain geometry since numerical diffusion is minimized in this situation.

Furthermore, quadrilateral/hexahedral elements permit a substantially larger aspect ratio without detrimental effect on the computational accuracy; thereby, allowing for a better solution with fewer cells in this situation as compared to a triangular/tetrahedral element mesh. Large aspect ratios invariably affect the skewness of triangular/tetrahedral cells, which typically hinders solution accuracy and convergence. As a result, despite the additional time and effort initially required to construct a domain mesh consisting of quadrilateral/hexahedral elements, the afforded benefits of reduced CPU clock time, increased rate of convergence, and decreased amount of numerical diffusion introduced into the solution are reasonably presumed to outweigh the initial costs associated with the flow domains modeled in this research. Several other noteworthy steps that are practically employed in reducing numerical diffusion include the use of a higher second-order discretization scheme, further mesh refinement (numerical diffusion is inversely related to grid resolution), and minimizing truncation errors that result from representing the fluid flow equations in discrete form. The latter of

these is subsequently addressed with issues pertaining to grid smoothness and overall mesh quality.

Attributes associated with the quality of a mesh are extremely influential toward the accuracy and stability of any numerical computation and include assessments of node point density/distribution, smoothness, and skewness. In other words, for the long flow domains of characteristically minute hydraulic diameters considered in this research, the mesh quality is highly dependent upon the orthogonality of the grid, nonuniform spacing and cell aspect ratios of grid elements, alignment of grid with the flow, and smoothness of the grid. Hence, the focus of mesh generation efforts (aside from the explicit mesh requirements formerly discussed) is directed expressly on these properties of the grid.

Orthogonal or near-orthogonal grids are desirable as the influence of cross-derivative terms in the discretized equations is minimized; thus, maintaining the skewness of grid elements within reasonably low limits is imperative. Nonuniform spacing and variable cell aspect ratios provide the potential for greater accuracy in domain regions where very large/steep gradients in the flow field exist without ensuing in unwieldy mesh sizes. However, expansion ratio limits exist and falling outside a moderate range of values results in a loss of accuracy. Due to discrete representation of a continuous domain, the degree to which relevant flow features are resolved depends unequivocally upon node/cell clustering within the volume region. In general, flow spaces necessitate a minimum of five cell elements and require a sufficiently fine mesh resolution in the region to minimize flow parameter differences between adjacent cells (FLUENT Inc., 2006).

As previously mentioned, alignment of grid elements with the flow further reduces numerical diffusion associated with the evaluation of advection terms, especially in lower-order differencing schemes. Rapid changes in cell volume as well as sizeable

variances in grid direction between a few adjacent cell elements compromises the smoothness of the grid, decreasing the accuracy of the interpolations applied by the FLUENT solver (*i.e.* increasing the difference between the partial derivatives in the governing equations and corresponding discrete approximations) and leading to larger truncation errors (Ball, 2004). Efforts to align the grid with the flow and preserve a reasonable gradient between differences in adjacent cell geometries are essential for construction of a truly high-quality mesh.

In summary, considerations for the general aspects of mesh generation outlined above are directly incorporated into the final mesh constructed for each of the three flow domain models. Although specific details are provided and discussed in Chapter 5 for each mesh generated, several commonalities exist among the final computational grids of all three models based solely on the aforementioned general properties and mesh quality desired. The approach utilized in generating each domain mesh is characterized as a multi-block method of sequential construction exploiting varying levels of flow volume decomposition. A multi-block mesh (as related to the current research models) simply describes a complete domain mesh that is comprised of two or more individual blocks of mesh for subvolume entities formed from decomposition of the flow domain volume. These blocks of mesh associated with the subvolumes are optionally stored as individual mesh files or any possible combination of grouped blocks of mesh including the potential collection of all blocks into a single base mesh file.

The multi-block meshes are further distinguished (within the current research models) by possessing interfaces between adjacent blocks of mesh that feature completely conformal mesh faces, which are subsequently fused in combining blocks of mesh saved within separate grid files to form the final domain mesh in the FLUENT software (if necessary). A multi-block mesh generation approach with decomposition of

the flow volume affords the benefit of simplifying a number of difficulties associated with the generation of a high quality quadrilateral/hexahedral mesh in complex domain geometries by sectioning the flow volume into smaller, manageable segments that are more conducive to representation by quadrilateral/hexahedral element shapes. Secondly, this method aids in circumventing the rather minute storage size limits for mesh files imposed by the crippling 32-bit platform constraints upon which the GAMBIT software database is built.

On a concluding note, the 'semi'-structured term of description is adopted from Tautges (n.d.) as a more appropriate label for the type of mesh that prevails throughout a majority of the embodied flow domain volumes; although, in a strict sense of the definition these meshes are of unstructured representation (and indeed the solver type in FLUENT as well). However, the final domain mesh generated for each model consists of two or more subvolume blocks of mesh with any combination of structured or unstructured grid representation, but limited to only Map, Submap, or Cooper scheme type options and quadrilateral/hexahedral cell elements. In other words, any unstructured subvolume block of mesh that is a constituent of a final domain mesh is generated from the Cooper volume meshing scheme type, resulting in a swept/extruded hexahedral mesh of the subvolume entity.

As highlighted by Tautges (n.d.), the swept/extruded hexahedral meshes are bounded in the next lower dimension by structured mesh and are consequently described with a 'semi'-structured aspect of consideration. Since each of the final computational domain meshes are formed by blocks of mesh with at least a semi-structured swept hexahedral mesh and more-often with a moderate number of structured blocks of mesh generated from Map/Submap scheme types, the semi-structured label is deemed reasonable for the domain meshes. Therefore, the final domain mesh for each of the

three numerical experiment models are generally denoted as multi-block semi-structured grids comprised entirely of quadrilateral/hexahedral cell elements.

Chapter 5: FLUENT CFD Simulation Trials and Results

5.1 Single Fully-Populated Fuel Bundle Run

Considering most phenomena in fluid mechanics depend in a complex fashion upon resulting parameter quantities that are both multifaceted and transcendental in relation, a primary initial task is the careful outlining of considerations for establishing a method in which the greatest quantity of accurate information is obtained in only a limited number of computational experiments. In the procession towards CFD modeling and simulation of a complete large-scale flow domain representing Sandia National Laboratories' prototypic GNF 9x9 BWR (GE 11 BWR/2-3) experimental fuel assembly, in which no prior experimental or computational data exists, the first objective is to conduct a number of initial computational analyses on smaller fuel assembly segments. This helps to establish a level of confidence in the correctness of the solver setup, boundary conditions, and mesh resolution of the flow domain in order to identify any potential problems or future obstacles that exist early in the development process before progressively modeling larger flow domains. The intent of the following computational simulations, results, and discussions is to address a number of these issues and contribute a minor collection of supplementary hydraulic analysis solutions in support of MELCOR SFP model calibration experiments conducted at SNL. Results from these preliminary numerical experiments are compared with the hydraulic analysis findings of SNL from the SFP pressure drop experiments.

The modeled flow domain in this first set of numerical case studies depicts a centered, two-thirds length span of a fully-populated fuel bundle run between two fully-populated spacer components. Dimensions and component surface constructs comprising

the flow domain segment are based on the accurate CAD solid model representation of the experimental apparatus at SNL, which is used directly in the development of the flow model in GAMBIT as previously outlined. Another illustration of the modeled flow domain is provided in Figure 5.1 below.

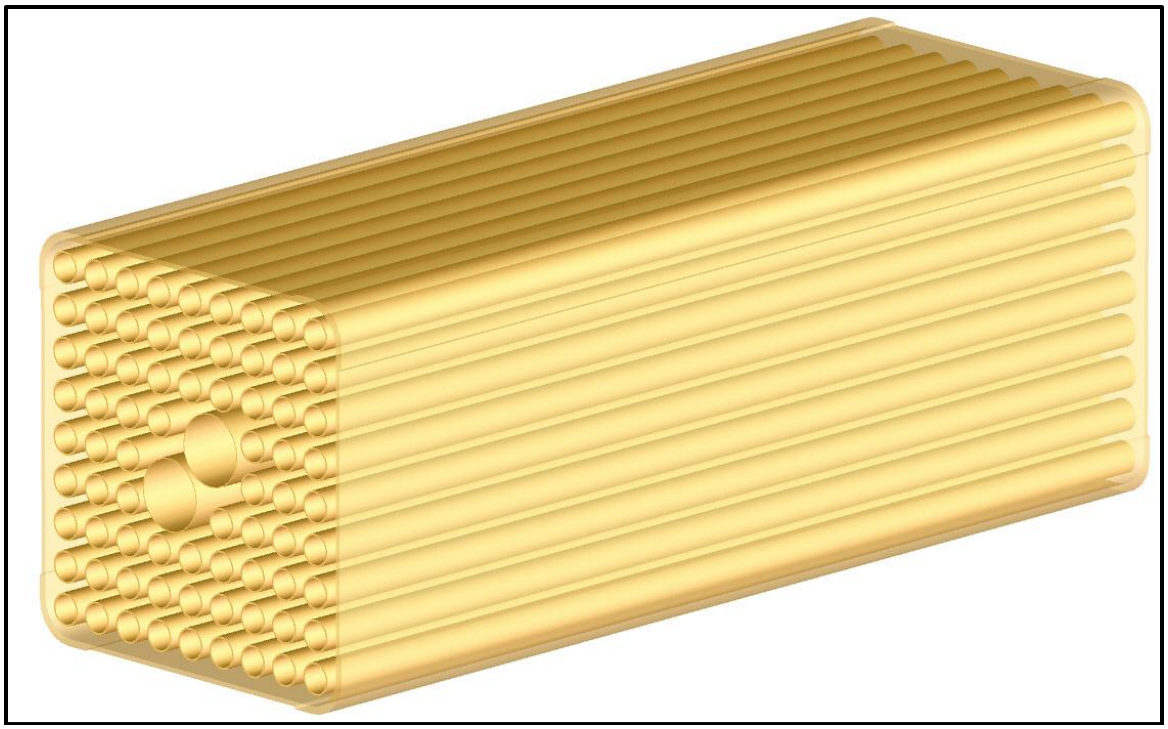


Figure 5.1: Modeled flow domain of a fully-populated fuel bundle run segment used in preliminary case study simulations.

The forced airflow rates examined through this fuel assembly segment varied in the range between approximately 100 and 500 standard liters per minute (slpm). A culmination of an estimated 2,000 CPU-hours is appropriated towards the completion of over 60 of the presumed most important simulation trials performed utilizing this CFD model. In an effort to outline these computational studies evaluating the relationship between characteristic hydraulic flow parameters and their impact on flow response of the modeled system, a test matrix summary is created which describes each of the

important analyses undertaken as well as model accuracy criterion or postulated flow behavior examined. The test matrix summary is shown in Figure 5.2 below.

TEST PLAN MATRIX OF COMPUTATIONAL EXPERIMENTS

STUDY:	PURPOSE:	FLOW-RATES:	PARAMETERS:	RESULTS:	NOTES:
[Preliminary Runs]	Verify problem is understood and properly setup; isolate any big concerns (*i.e.* invalid grid)	Randomly selected between ~100 – 500 slpm for mass flow rate	Pressure loss (ΔP), velocity (V_{out}), and wall (Y^+) values	Numerical tabulations included with some discussion	Get a feel for problem; resources, time, mesh sizes needed, also with varied inlet-outlet BCs
[Grid-Independence]	Access truncation error to show convergence, order of accuracy, and model correctness	~150, 450 slpm	ΔP (integral quantity), Error (relative)	Relative error in pressure loss compared to finest mesh result with discussion	Grids range from ~ 1 to > 7 million node points
[Periodicity & Symmetry Check]	Check symmetry is enforced, analyze fully developed flow	~150, 450 slpm	Pressure loss (ΔP), velocity ($V_{in/out}$), and velocity profiles with contour plots	Numerical and graphical, mostly discussed; compare other model flow developments in final results	Vital information needed for contour difference plots compared in final results
[Viscous Models]	Determine and validate viscous models and options	Randomly selected between ~100 – 500 slpm	ΔP, fluid flow observations, models are choices	Numerically tabulated with some discussion	Mesh is critical to accuracy of model chosen
[Boundary Conditions & Turbulence Parameters]	Check for application correctness, type of system response, model BC fidelity	At flow rates previously examined	ΔP, max $V_{in/out}$, max (Y^+), pressures (*i.e.* static, total)…+	Numerically tabulated, a few graphics, and brief discussion	Considered together since all inlets and outlets with BCs require inputs for both
[Residual Criteria]	Verify point of convergence accurately	~150, 450 slpm	ΔP, V_{out}	Numerically tabulated	
[Convergence Monitors]	Introduce available criterion options	At flow rates previously examined	Variable	Discussion of several different monitors implemented throughout studies	
[Final Results]	Based on all computational experiments and case studies performed above, these are determined to be the best and final results! (Selected presentation of graphical outputs, compared to experimental results where applicable)				

Figure 5.2: Test plan matrix summarizing important preliminary computational studies performed.

Although a wide range of variable modeling parameters and flow solver options are incorporated into the test matrix in Figure 5.2, there are a few general characteristics and assumptions applicable to all models employed and studies conducted in this test matrix. These basic modeling details and solver setup options are introduced below.

Common Model Details and Assumptions:

- The mesh model of the flow domain is comprised entirely of quadrilateral/hexahedral cell elements, the characteristics of which are provided along with model geometry details in the numerical results tabulated for each analysis.
- The fluid airflow is incompressible (constant density) with property values taken as the average local ambient conditions reported in the hydraulic analysis summary by Durbin and Lindgren (2005) for the SFP pressure drop experiments performed at SNL.
- Applied wall surface boundary conditions include adiabatic, no-slip, negligible viscous heating, and default roughness.
- Absence of sufficient inlet boundary profile information necessitates the use of a uniform or periodic inlet profile unless otherwise noted.
- In simulations utilizing turbulence viscosity models, the turbulent boundary condition parameters at inlets and outlets are specified using the hydraulic diameter and turbulence intensity method; the hydraulic diameter is already known and the intensity is initially approximated using appropriate formulae from Chapter 7 of the FLUENT user's guide (FLUENT Inc., 2006).
- Unless otherwise noted, the convergence criterion for scaled residuals is at least $1x10^{-4}$ for all equations, but set tighter (*i.e.* $1x10^{-5}$) in several simulation studies.

- If not explicitly stated in accompanied descriptions of modeling options implemented and assumptions imposed for each analysis, assume default values.

5.1.1 Preliminary Case Trials

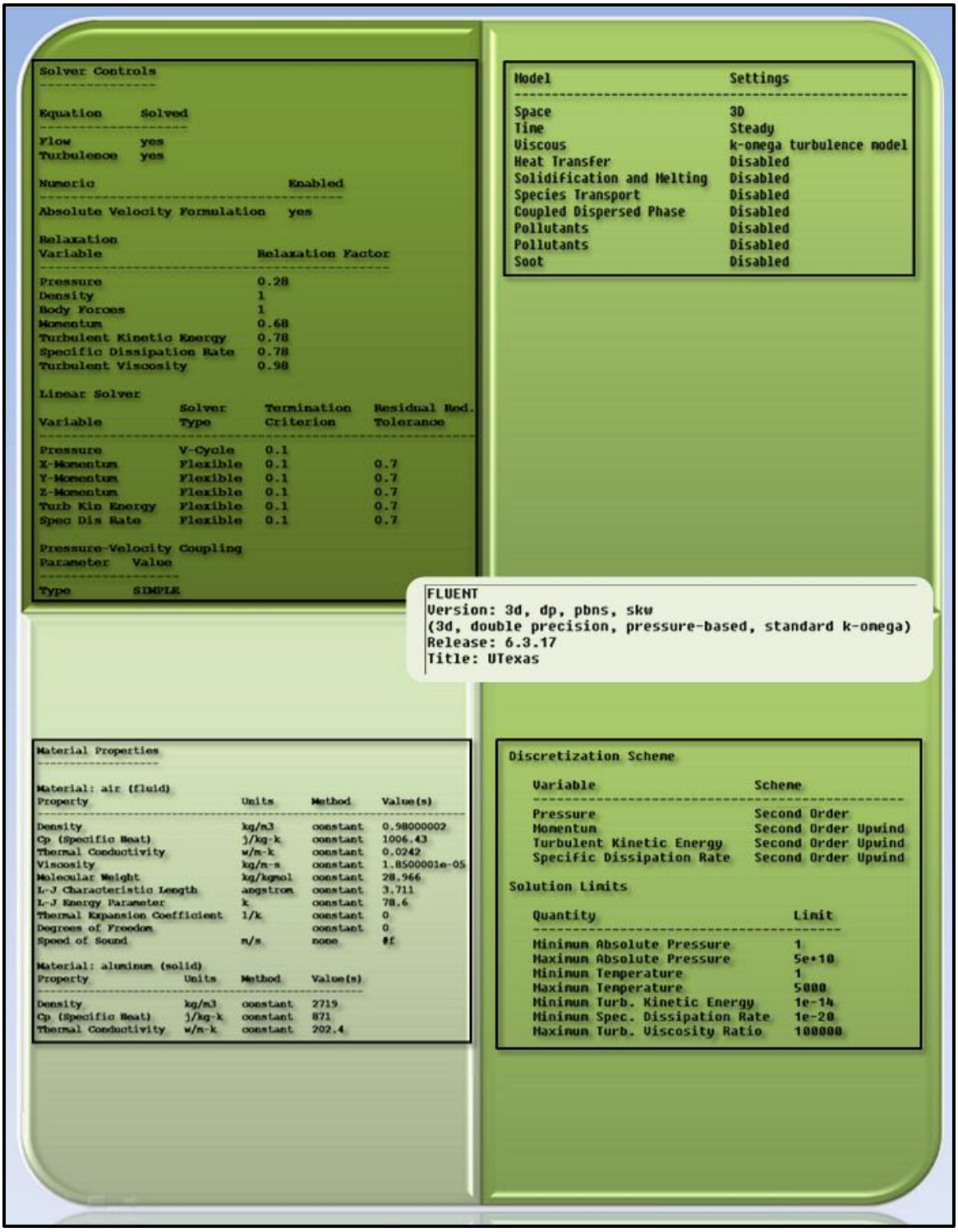
The impetus for conducting the 'preliminary base trial runs' is merely to gain an initial impression of the overall features of the flow field solution, quickly verify that the problem is understood through correct translation of problem description to flow domain representation, and initially gauge the computational resource requirements in relation to wall-clock time per simulation. As many guideline parameters for establishing proper mesh resolution and solver setup are directly dependent on the flow-field solution requiring *a priori* knowledge of the simulation results, other methods of a more systematic nature are utilized to arrive at acceptable solutions. Therefore, a vast majority of modeling decisions at this point are based on user experience from previous analyses involving similar flows along with best case estimates for gauging the relevance of a turbulence model (perhaps based on Reynolds number), dimension of the first grid point off bounding walls (y^+) with appropriate wall treatment, and other applicable guidelines known.

The key fundamentals and preliminary modeling assumptions implemented in each of these initial case studies resurface for debate with the final results and conclusions section that follows the subsequent discussions. As a result, it simply suffices to note a few important setup characteristics attributed to these initial trial runs, which are ultimately selected as most appropriate at the time of execution.

Key Points of Preliminary Trial Runs:

- The standard k-ω (transitional flow option) viscous model is selected for initial study despite relatively low-Re flow expectations (laminar flow regime) due to presence of bounding walls and uncertainty of inlet conditions.
- Mesh generated from past experience and estimated maximum (y) value from Equations (2.74) and (2.75) with (y^+) equal to one, since enhanced wall treatments are intended for application in order to accurately measure pressure loss in wall-bounded flow. A minimum of 10 cell elements are required between all adjacent boundary wall surfaces.
- A uniform inlet boundary profile is imposed for these initial trial runs.
- The inlet and outlet boundary conditions are randomly chosen within the realm of permissible options and monitored at ~ 450 slpm for each boundary condition set (for comparison) as well as individually at one additional flow rate (to expand the range of values investigated).

In addition to the general information above, Figures 5.3, 5.4, and 5.5 provide a complete outline of all modeling setup parameters for the velocity inlet - pressure outlet simulation at ~ 450 slpm. This information is imparted as an example of the typical trial setup and listing of considerations that are taken into account.



Solver Controls

Equation	Solved
Flow	yes
Turbulence	yes

Numeric	Enabled
Absolute Velocity Formulation	yes

Relaxation

Variable	Relaxation Factor
Pressure	0.28
Density	1
Body Forces	1
Momentum	0.68
Turbulent Kinetic Energy	0.78
Specific Dissipation Rate	0.78
Turbulent Viscosity	0.98

Linear Solver

Variable	Solver Type	Termination Criterion	Residual Red. Tolerance
Pressure	V-Cycle	0.1	
X-Momentum	Flexible	0.1	0.7
Y-Momentum	Flexible	0.1	0.7
Z-Momentum	Flexible	0.1	0.7
Turb Kin Energy	Flexible	0.1	0.7
Spec Dis Rate	Flexible	0.1	0.7

Pressure-Velocity Coupling

Parameter	Value
Type	SIMPLE

Model	Settings
Space	3D
Time	Steady
Viscous	k-omega turbulence model
Heat Transfer	Disabled
Solidification and Melting	Disabled
Species Transport	Disabled
Coupled Dispersed Phase	Disabled
Pollutants	Disabled
Pollutants	Disabled
Soot	Disabled

FLUENT
Version: 3d, dp, pbns, sku
(3d, double precision, pressure-based, standard k-omega)
Release: 6.3.17
Title: UTexas

Material Properties

Material: air (fluid)

Property	Units	Method	Value(s)
Density	kg/m3	constant	0.98000002
Cp (Specific Heat)	j/kg-k	constant	1006.43
Thermal Conductivity	w/m-k	constant	0.0242
Viscosity	kg/m-s	constant	1.8500001e-05
Molecular Weight	kg/kgmol	constant	28.966
L-J Characteristic Length	angstrom	constant	3.711
L-J Energy Parameter	k	constant	78.6
Thermal Expansion Coefficient	1/k	constant	0
Degrees of Freedom		constant	0
Speed of Sound	m/s	none	#f

Material: aluminum (solid)

Property	Units	Method	Value(s)
Density	kg/m3	constant	2719
Cp (Specific Heat)	j/kg-k	constant	871
Thermal Conductivity	w/m-k	constant	202.4

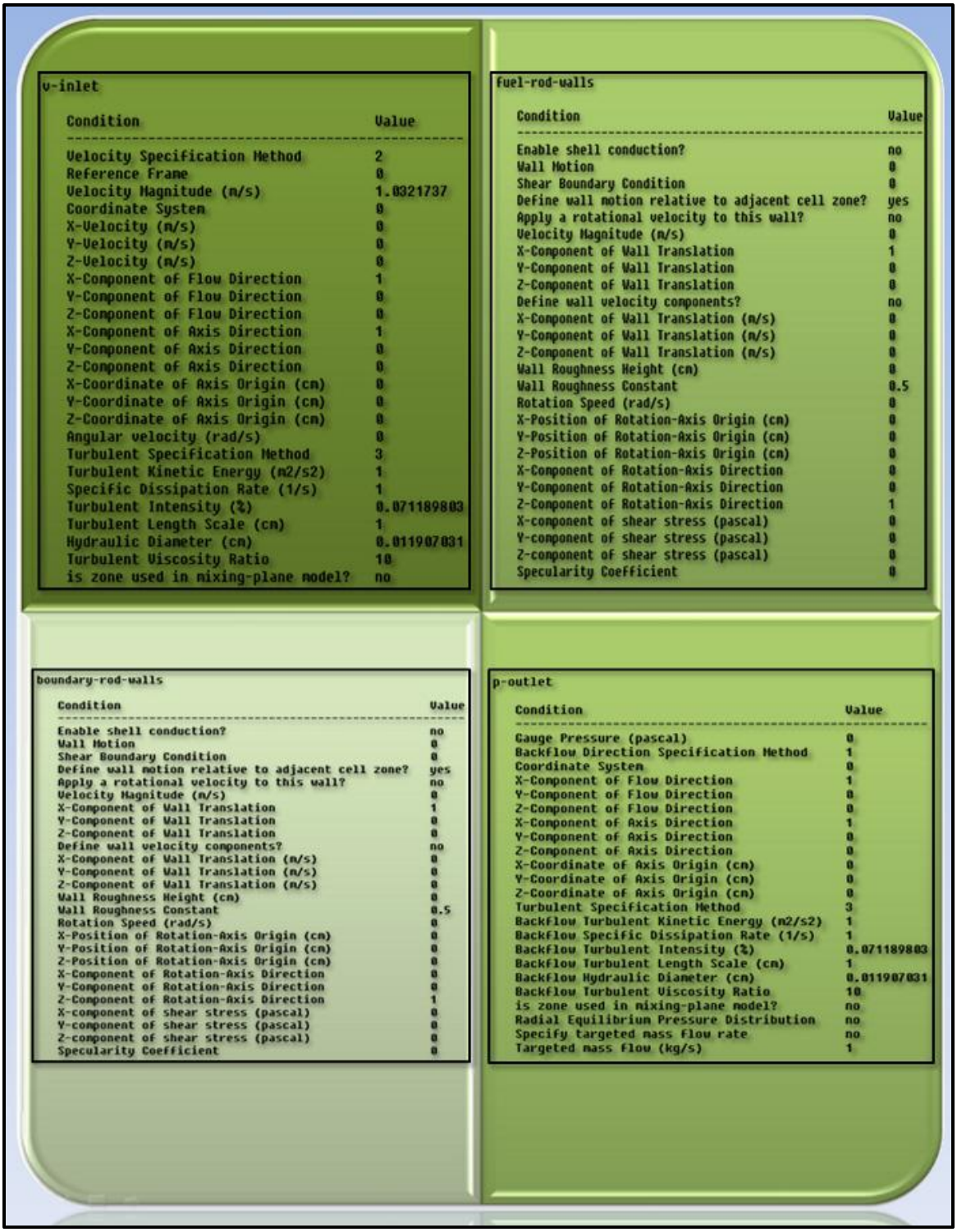
Discretization Scheme

Variable	Scheme
Pressure	Second Order
Momentum	Second Order Upwind
Turbulent Kinetic Energy	Second Order Upwind
Specific Dissipation Rate	Second Order Upwind

Solution Limits

Quantity	Limit
Minimum Absolute Pressure	1
Maximum Absolute Pressure	5e+10
Minimum Temperature	1
Maximum Temperature	5000
Minimum Turb. Kinetic Energy	1e-14
Minimum Spec. Dissipation Rate	1e-20
Maximum Turb. Viscosity Ratio	100000

Figure 5.3: Report summary for velocity inlet - pressure outlet at ~ 450 slpm (1 of 3).



v-inlet

Condition	Value
Velocity Specification Method	2
Reference Frame	0
Velocity Magnitude (m/s)	1.0321737
Coordinate System	0
X-Velocity (m/s)	0
Y-Velocity (m/s)	0
Z-Velocity (m/s)	0
X-Component of Flow Direction	1
Y-Component of Flow Direction	0
Z-Component of Flow Direction	0
X-Component of Axis Direction	1
Y-Component of Axis Direction	0
Z-Component of Axis Direction	0
X-Coordinate of Axis Origin (cm)	0
Y-Coordinate of Axis Origin (cm)	0
Z-Coordinate of Axis Origin (cm)	0
Angular velocity (rad/s)	0
Turbulent Specification Method	3
Turbulent Kinetic Energy (m2/s2)	1
Specific Dissipation Rate (1/s)	1
Turbulent Intensity (%)	0.071189803
Turbulent Length Scale (cm)	1
Hydraulic Diameter (cm)	0.011907031
Turbulent Viscosity Ratio	10
is zone used in mixing-plane model?	no

fuel-rod-walls

Condition	Value
Enable shell conduction?	no
Wall Motion	0
Shear Boundary Condition	0
Define wall motion relative to adjacent cell zone?	yes
Apply a rotational velocity to this wall?	no
Velocity Magnitude (m/s)	0
X-Component of Wall Translation	1
Y-Component of Wall Translation	0
Z-Component of Wall Translation	0
Define wall velocity components?	no
X-Component of Wall Translation (m/s)	0
Y-Component of Wall Translation (m/s)	0
Z-Component of Wall Translation (m/s)	0
Wall Roughness Height (cm)	0
Wall Roughness Constant	0.5
Rotation Speed (rad/s)	0
X-Position of Rotation-Axis Origin (cm)	0
Y-Position of Rotation-Axis Origin (cm)	0
Z-Position of Rotation-Axis Origin (cm)	0
X-Component of Rotation-Axis Direction	0
Y-Component of Rotation-Axis Direction	0
Z-Component of Rotation-Axis Direction	1
X-component of shear stress (pascal)	0
Y-component of shear stress (pascal)	0
Z-component of shear stress (pascal)	0
Specularity Coefficient	0

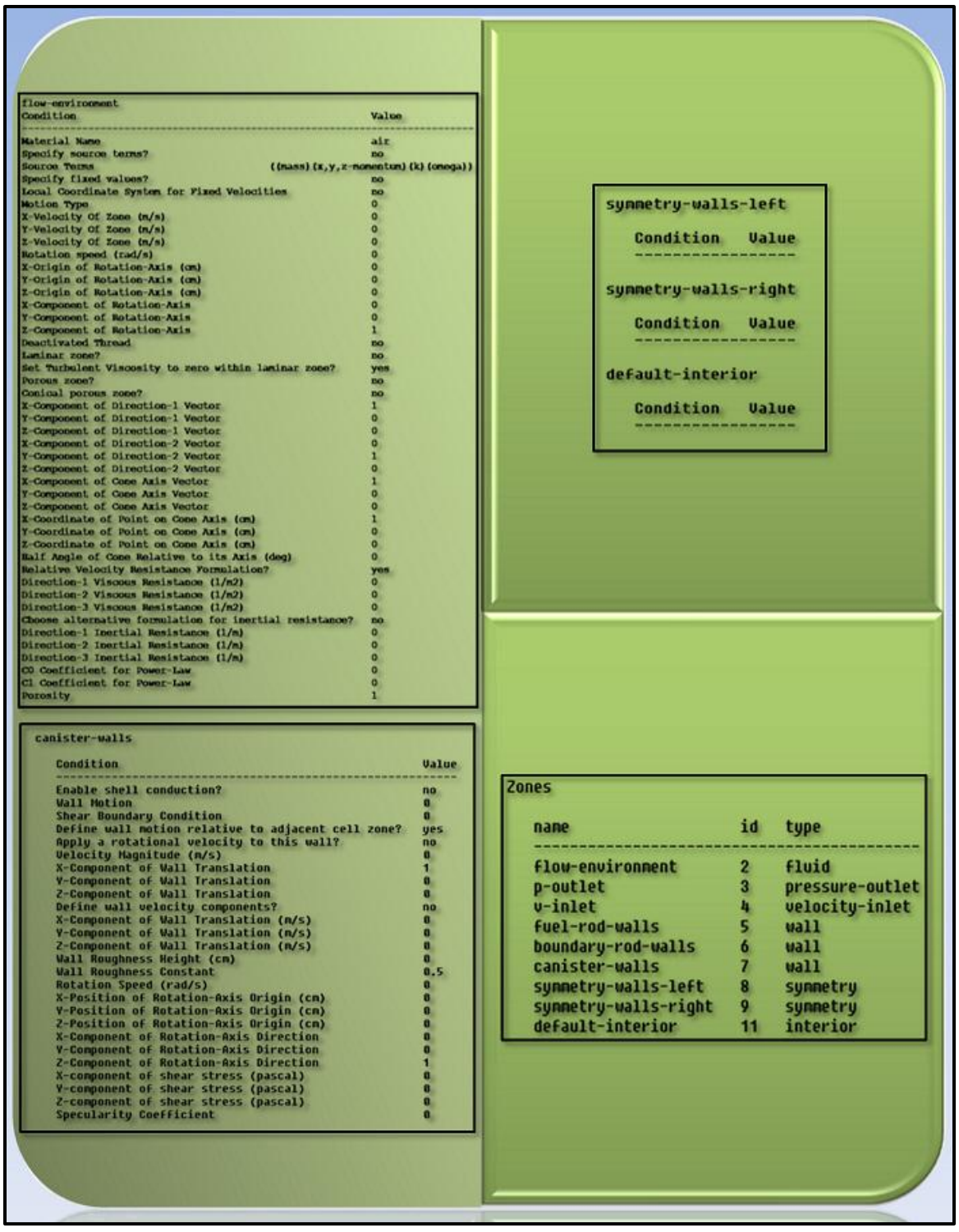
boundary-rod-walls

Condition	Value
Enable shell conduction?	no
Wall Motion	0
Shear Boundary Condition	0
Define wall motion relative to adjacent cell zone?	yes
Apply a rotational velocity to this wall?	no
Velocity Magnitude (m/s)	0
X-Component of Wall Translation	1
Y-Component of Wall Translation	0
Z-Component of Wall Translation	0
Define wall velocity components?	no
X-Component of Wall Translation (m/s)	0
Y-Component of Wall Translation (m/s)	0
Z-Component of Wall Translation (m/s)	0
Wall Roughness Height (cm)	0
Wall Roughness Constant	0.5
Rotation Speed (rad/s)	0
X-Position of Rotation-Axis Origin (cm)	0
Y-Position of Rotation-Axis Origin (cm)	0
Z-Position of Rotation-Axis Origin (cm)	0
X-Component of Rotation-Axis Direction	0
Y-Component of Rotation-Axis Direction	0
Z-Component of Rotation-Axis Direction	1
X-component of shear stress (pascal)	0
Y-component of shear stress (pascal)	0
Z-component of shear stress (pascal)	0
Specularity Coefficient	0

p-outlet

Condition	Value
Gauge Pressure (pascal)	0
Backflow Direction Specification Method	1
Coordinate System	0
X-Component of Flow Direction	1
Y-Component of Flow Direction	0
Z-Component of Flow Direction	0
X-Component of Axis Direction	1
Y-Component of Axis Direction	0
Z-Component of Axis Direction	0
X-Coordinate of Axis Origin (cm)	0
Y-Coordinate of Axis Origin (cm)	0
Z-Coordinate of Axis Origin (cm)	0
Turbulent Specification Method	3
Backflow Turbulent Kinetic Energy (m2/s2)	1
Backflow Specific Dissipation Rate (1/s)	1
Backflow Turbulent Intensity (%)	0.071189803
Backflow Turbulent Length Scale (cm)	1
Backflow Hydraulic Diameter (cm)	0.011907031
Backflow Turbulent Viscosity Ratio	10
is zone used in mixing-plane model?	no
Radial Equilibrium Pressure Distribution	no
Specify targeted mass flow rate	no
Targeted mass flow (kg/s)	1

Figure 5.4: Report summary for velocity inlet - pressure outlet at ~ 450 slpm (2 of 3).



flow-environment

Condition	Value
Material Name	air
Specify source terms?	no
Source Terms	((mass) (x,y,z-momentum) (k) (omega))
Specify fixed values?	no
Local Coordinate System for Fixed Velocities	no
Motion Type	0
X-Velocity Of Zone (m/s)	0
Y-Velocity Of Zone (m/s)	0
Z-Velocity Of Zone (m/s)	0
Rotation speed (rad/s)	0
X-Origin of Rotation-Axis (cm)	0
Y-Origin of Rotation-Axis (cm)	0
Z-Origin of Rotation-Axis (cm)	0
X-Component of Rotation-Axis	0
Y-Component of Rotation-Axis	0
Z-Component of Rotation-Axis	1
Deactivated Thread	no
Laminar zone?	no
Set Turbulent Viscosity to zero within laminar zone?	yes
Porous zone?	no
Conical porous zone?	no
X-Component of Direction-1 Vector	1
Y-Component of Direction-1 Vector	0
Z-Component of Direction-1 Vector	0
X-Component of Direction-2 Vector	0
Y-Component of Direction-2 Vector	1
Z-Component of Direction-2 Vector	0
X-Component of Cone Axis Vector	1
Y-Component of Cone Axis Vector	0
Z-Component of Cone Axis Vector	0
X-Coordinate of Point on Cone Axis (cm)	1
Y-Coordinate of Point on Cone Axis (cm)	0
Z-Coordinate of Point on Cone Axis (cm)	0
Half Angle of Cone Relative to its Axis (deg)	0
Relative Velocity Resistance Formulation?	yes
Direction-1 Viscous Resistance (1/m2)	0
Direction-2 Viscous Resistance (1/m2)	0
Direction-3 Viscous Resistance (1/m2)	0
Choose alternative formulation for inertial resistance?	no
Direction-1 Inertial Resistance (1/m)	0
Direction-2 Inertial Resistance (1/m)	0
Direction-3 Inertial Resistance (1/m)	0
C0 Coefficient for Power-Law	0
C1 Coefficient for Power-Law	0
Porosity	1

canister-walls

Condition	Value
Enable shell conduction?	no
Wall Motion	0
Shear Boundary Condition	0
Define wall motion relative to adjacent cell zone?	yes
Apply a rotational velocity to this wall?	no
Velocity Magnitude (m/s)	0
X-Component of Wall Translation	1
Y-Component of Wall Translation	0
Z-Component of Wall Translation	0
Define wall velocity components?	no
X-Component of Wall Translation (m/s)	0
Y-Component of Wall Translation (m/s)	0
Z-Component of Wall Translation (m/s)	0
Wall Roughness Height (cm)	0
Wall Roughness Constant	0.5
Rotation Speed (rad/s)	0
X-Position of Rotation-Axis Origin (cm)	0
Y-Position of Rotation-Axis Origin (cm)	0
Z-Position of Rotation-Axis Origin (cm)	0
X-Component of Rotation-Axis Direction	0
Y-Component of Rotation-Axis Direction	0
Z-Component of Rotation-Axis Direction	1
X-component of shear stress (pascal)	0
Y-component of shear stress (pascal)	0
Z-component of shear stress (pascal)	0
Specularity Coefficient	0

symmetry-walls-left

Condition	Value

symmetry-walls-right

Condition	Value

default-interior

Condition	Value

Zones

name	id	type
flow-environment	2	fluid
p-outlet	3	pressure-outlet
v-inlet	4	velocity-inlet
fuel-rod-walls	5	wall
boundary-rod-walls	6	wall
canister-walls	7	wall
symmetry-walls-left	8	symmetry
symmetry-walls-right	9	symmetry
default-interior	11	interior

Figure 5.5: Report summary for velocity inlet - pressure outlet at ~ 450 slpm (3 of 3).

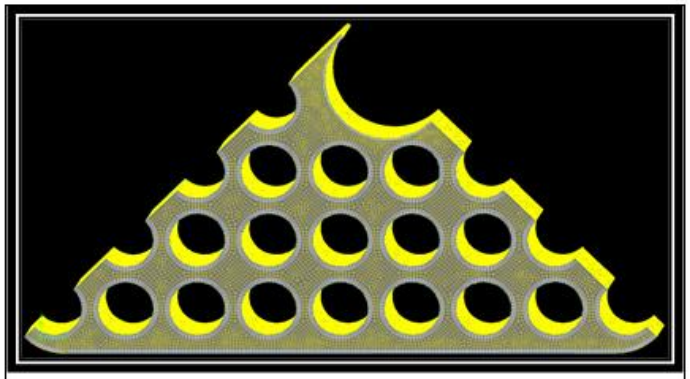
Apart from the more broadly stated purpose of targeting pertinent features of typical flow field solutions with reasonable detail and verifying the first model grid is correctly fabricated in order to resolve these features, there are several more specific quantitative values of interest. The primary parameter quantities are outlined in the test plan matrix illustrated in Figure 5.2 above, which include pressure losses across the modeled fuel bundle run segment, maximum outlet flow velocities, and (y^+) values at boundary walls. Qualitative observations of velocity and pressure profile developments along the axial length of the flow domain are also of considerable importance.

The numerical and postprocessed graphical solutions from six of the preliminary testing experiments are selected based upon a reasonable sampling of the entire range of flow rates and solver setups evaluated for characteristic flow-field values of designated interest. Although presenting the results from all simulations is unfeasible, the following tabulated numerical values along with postprocessed graphical flow fields in Figures 5.6 through 5.11 below provide sufficient resulting data to formulate numerous inferences with regard to the initial grid resolution and general features of the flow field. This information is invaluable in providing an effective overview of characteristics inherent to the general flow domain and a relatively accurate point of reference to gauge starting mesh resolutions for the grid refinement process.

SETUP CONDITIONS AND RESULTS FOR PRELIMINARY RUNS AT SUBJECTIVE FLOW RATES

STATIC VALUES

(Quantity or Data of Interest):	(Value):
Air Density (kg/m^3)	0.98
Air Viscosity (kg/s-m) $_{dynamic}$	1.85E-05
Flow Area (m^2) $_{x\text{-}section}$	0.00968829
Wetted Perimeter (m) $_{x\text{-}section}$	3.254596
Hydraulic Diameter (m)	0.011907
Turbulence Model	k-w standard $_{transitional}$
Face Cell Count $_{x\text{-}section}$	8667
Total Volume Cell Count $_{domain}$	2773440
Total Node Count $_{domain}$	3030882
Total Model Length (m) $_{domain}$	0.32088667



Meshed cross-section of flow domain used with ~ 3 million total nodes.

BCs AND RESULTS - VELOCITY INLET & PRESSURE OUTLET

FLOW RATE ~ 450 (SLPM)

BC Variable:	Value:
Domain Flow (standard liters per minute)	454
Air Velocity (m/s) $_{inlet}$	1.032173701
Turbulence Intensity (%) $_{inlet}$	7.1189805
Gauge Pressure (Pa) $_{outlet}$	0
Turbulence Intensity (%) $_{outlet}$	7.1189805
Results	
Static Pressure Difference (Pa)	2.6612415
Maximum Velocity (m/s) $_{outlet}$	2.2814283
Maximum Y-Plus Value	2.7277656
Average Y-Plus Value	0.17543125

FLOW RATE ~ 300 (SLPM)

BC Variable:	Value:
Domain Flow (standard liters per minute)	303
Air Velocity (m/s) $_{inlet}$	0.6881158
Turbulence Intensity (%) $_{inlet}$	3
Gauge Pressure (Pa) $_{outlet}$	0
Turbulence Intensity (%) $_{outlet}$	0.3
Results	
Static Pressure Difference (Pa)	1.6453676
Maximum Velocity (m/s) $_{outlet}$	1.6572793
Maximum Y-Plus Value	2.1592987
Average Y-Plus Value	0.14002419

Figure 5.6: Selected numerical results from preliminary case trial runs (1 of 2).

BCs AND RESULTS - MASS FLOW INLET & PRESSURE OUTLET

FLOW RATE	~ 450 (SLPM)	
	BC Variable:	Value:
	Domain Flow (standard liters per minute)	454
	Air Mass Flow Rate (kg/s) $_{inlet}$	0.0098
	Turbulence Intensity (%) $_{inlet}$	7.1189805
	Gauge Pressure (Pa) $_{outlet}$	0
	Turbulence Intensity (%) $_{outlet}$	7.1189805
	Results	
	Static Pressure Difference (Pa)	2.6569006
	Maximum Velocity (m/s) $_{outlet}$	2.2790601
	Maximum Y-Plus Value	2.7256086
	Average Y-Plus Value	0.1752973

FLOW RATE	~ 230 (SLPM)	
	BC Variable:	Value:
	Domain Flow (standard liters per minute)	227
	Air Mass Flow Rate (kg/s) $_{inlet}$	0.0049
	Turbulence Intensity (%) $_{inlet}$	1
	Gauge Pressure (Pa) $_{outlet}$	0
	Turbulence Intensity (%) $_{outlet}$	0.1
	Results	
	Static Pressure Difference (Pa)	1.1770593
	Maximum Velocity (m/s) $_{outlet}$	1.3073766
	Maximum Y-Plus Value	1.8290533
	Average Y-Plus Value	0.1195567

BCs AND RESULTS - MASS FLOW INLET & OUTFLOW OUTLET

FLOW RATE	~ 450 (SLPM)	
	BC Variable:	Value:
	Domain Flow (standard liters per minute)	454
	Air Mass Flow Rate (kg/s) $_{inlet}$	0.0098
	Turbulence Intensity (%) $_{inlet}$	7.1189805
	Flow Rate Weighting $_{outlet}$	1
	Results	
	Static Pressure Difference (Pa)	2.6594483
	Maximum Velocity (m/s) $_{outlet}$	2.2918265
	Maximum Y-Plus Value	2.7256086
	Average Y-Plus Value	0.17528076

FLOW RATE	~ 150 (SLPM)	
	BC Variable:	Value:
	Domain Flow (standard liters per minute)	151
	Air Mass Flow Rate (kg/s) $_{inlet}$	0.003266667
	Turbulence Intensity (%) $_{inlet}$	0.1
	Flow Rate Weighting $_{outlet}$	1
	Results	
	Static Pressure Difference (Pa)	0.74445092
	Maximum Velocity (m/s) $_{outlet}$	0.94011164
	Maximum Y-Plus Value	1.4476023
	Average Y-Plus Value	0.09609689

Figure 5.7: Selected numerical results from preliminary case trial runs (2 of 2).

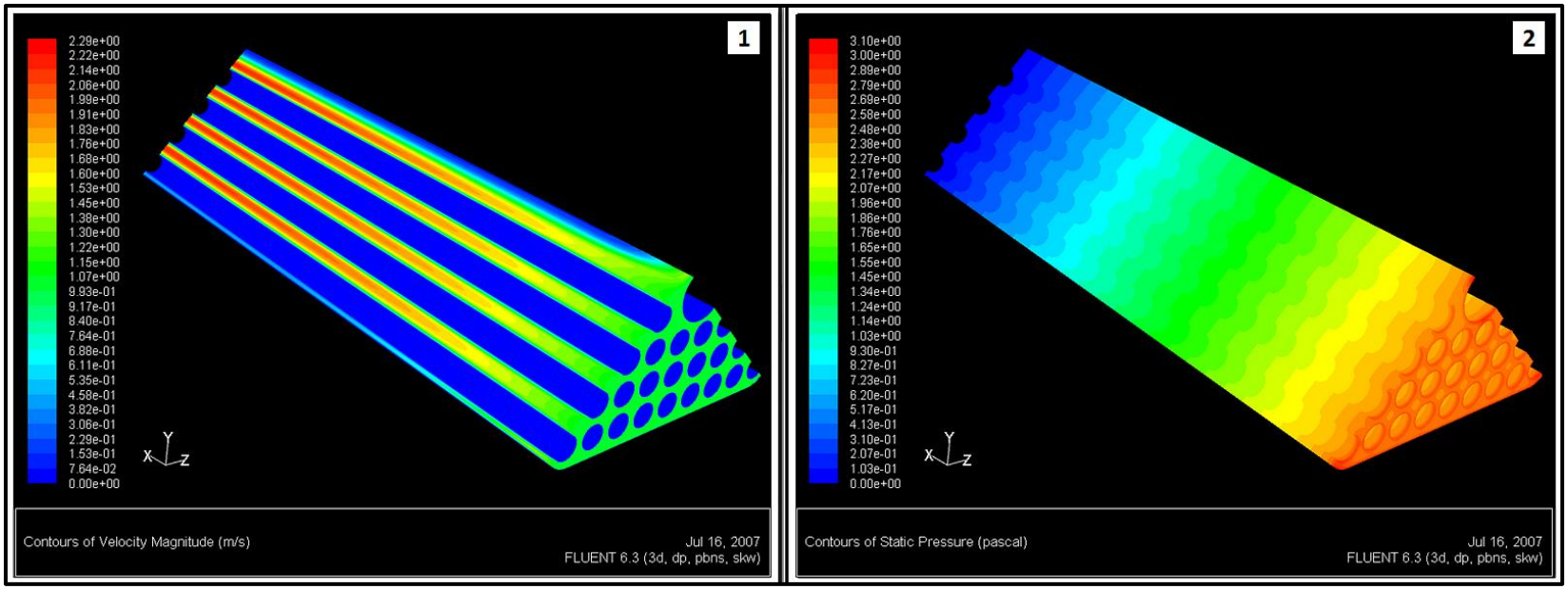


Figure 5.8: Contour plot results from preliminary trial run at ~ 450 slpm with uniform velocity inlet [near] and pressure outlet [far]; (1) velocity magnitude, (2) static pressure.

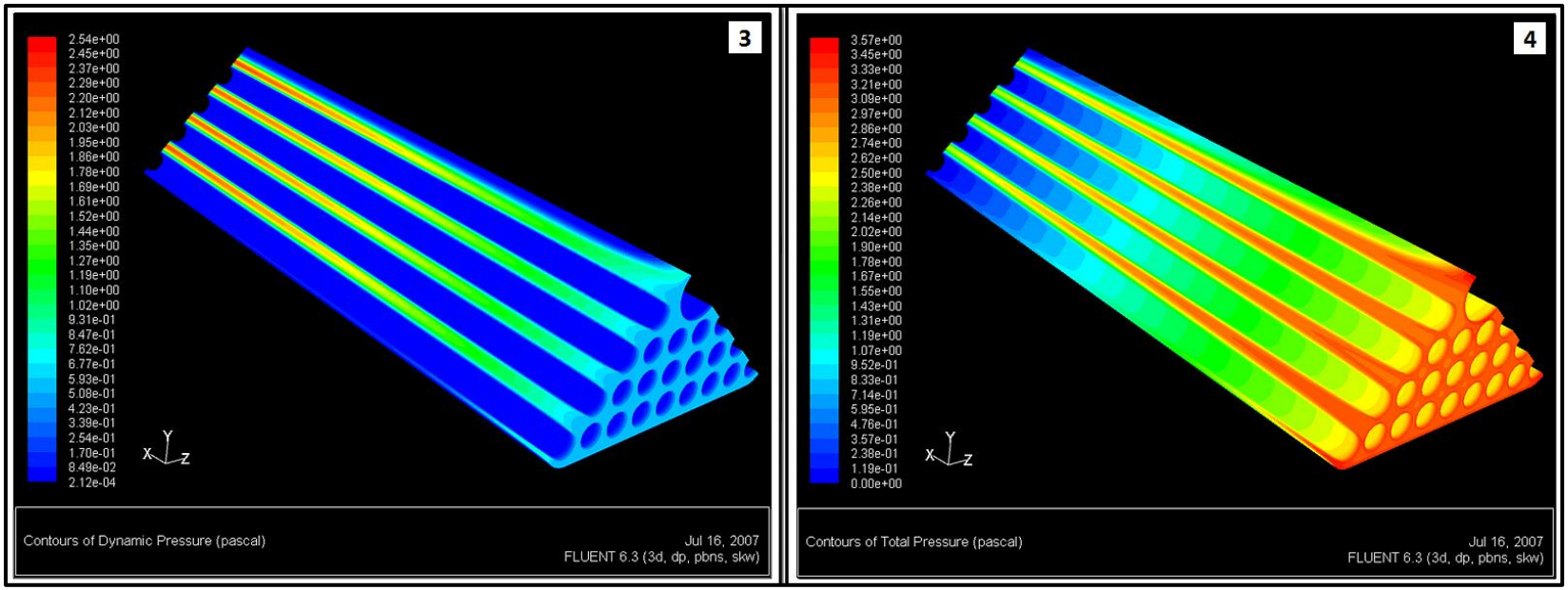


Figure 5.9: Contour plot results from preliminary trial run at ~ 450 slpm with uniform velocity inlet [near] and pressure outlet [far]; (3) dynamic pressure, (4) total pressure.

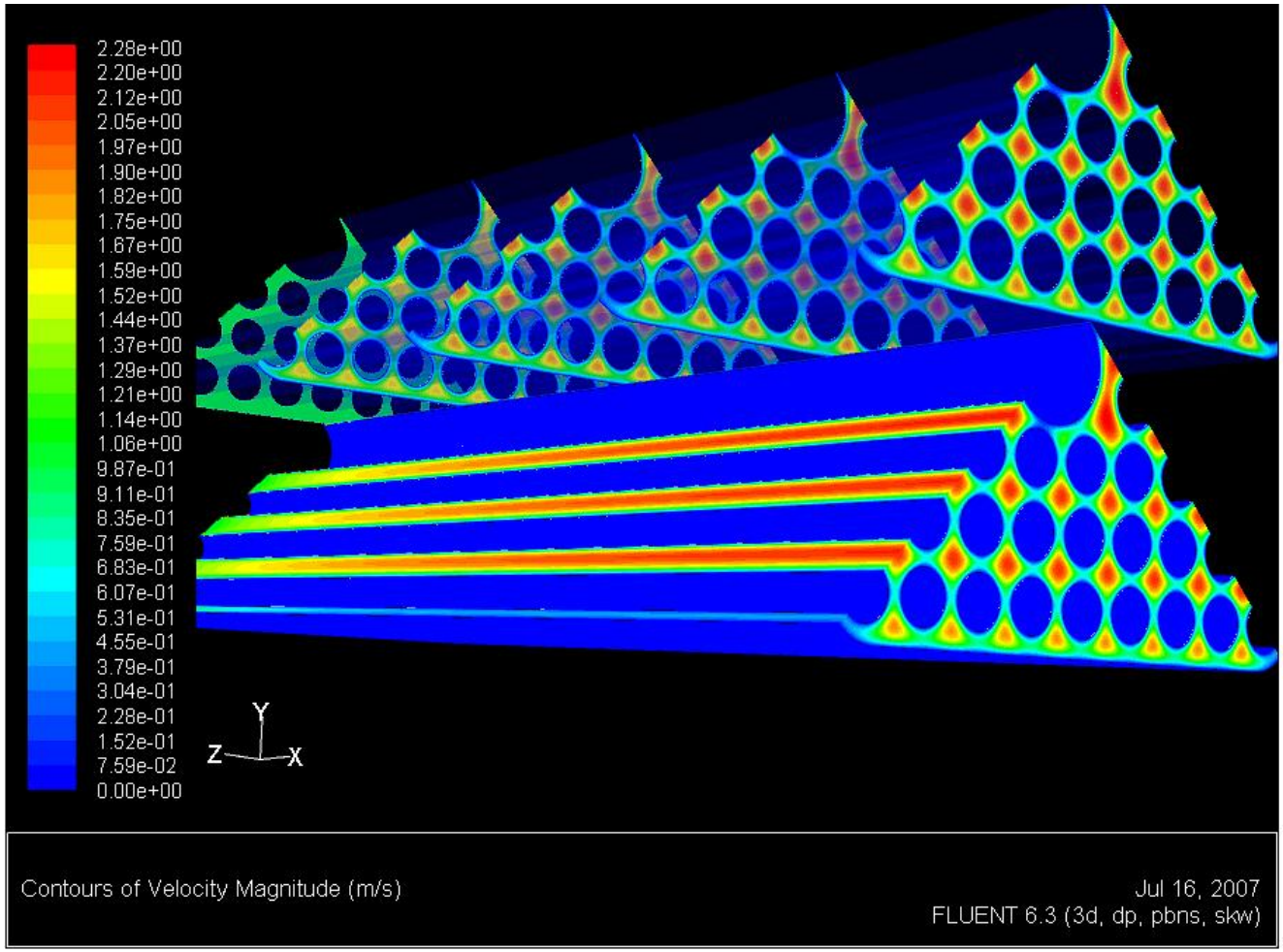


Figure 5.10: Contour plot (1) from Figure 5.8 with pressure outlet [near] and additional cross-section contours.

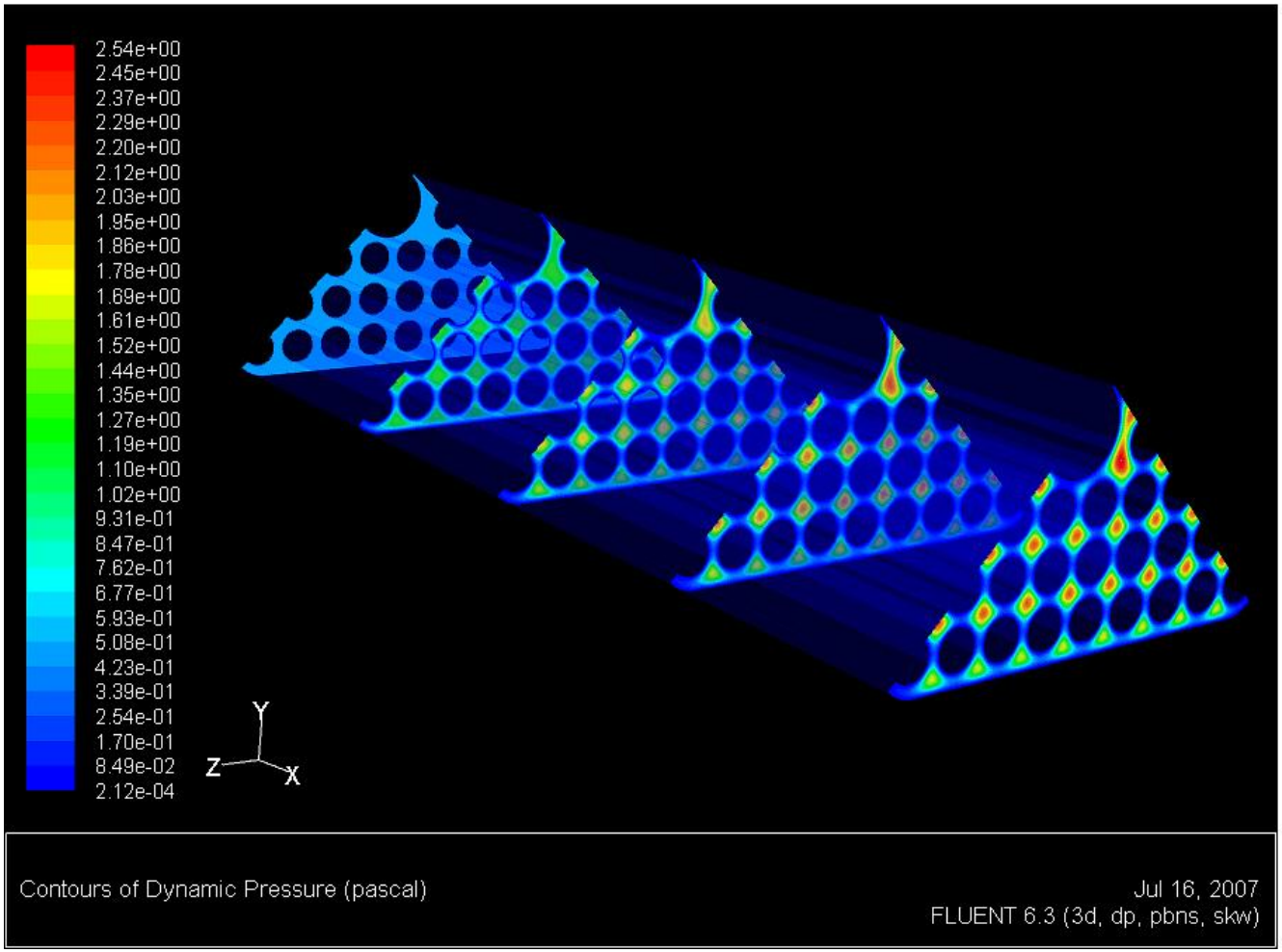


Figure 5.11: Contour plot (3) from Figure 5.9 with pressure outlet [near] and only highlighted cross-section contours.

In addition, due to the importance of varying profile developments throughout the domain, the deviations in associated quantities in the flow direction, and the lack of initial inlet/outlet values for most parameters, line/rake surfaces are created at various cross-section locations along the length of the domain to analyze a limited number of evolving characteristics in several trials performed. Reference diagrams with construction details and profile results for a set of line surfaces employed during numerous postprocessing efforts of simulations conducted are provided in Figures 5.12 through 5.15 below. These data points are carefully located at positions within the flow field (or cross-sectional area of flow) thought to contain critical characteristic behaviors (or trends) defining the general flow response of the system within a reasonable vicinity of cross-sectional cells.

As a result, by analyzing different groups of these profile data points (line/rake surfaces) at cross-sections along the domain length, it is possible to associate certain variable values and trends in flow-field data to resulting overall system response. Again, the following figures represent only a select fraction of the comprehensive profile sets evaluated and only for a single parameter of particular interest, but provide sufficient information to introduce the method by which a number of final results are reported in the concluding discussion section. The results below are taken from preliminary trial runs at airflow rates of ~ 450 and 150 slpm with a mass flow boundary type at the inlet and an outflow boundary type at the outlet.

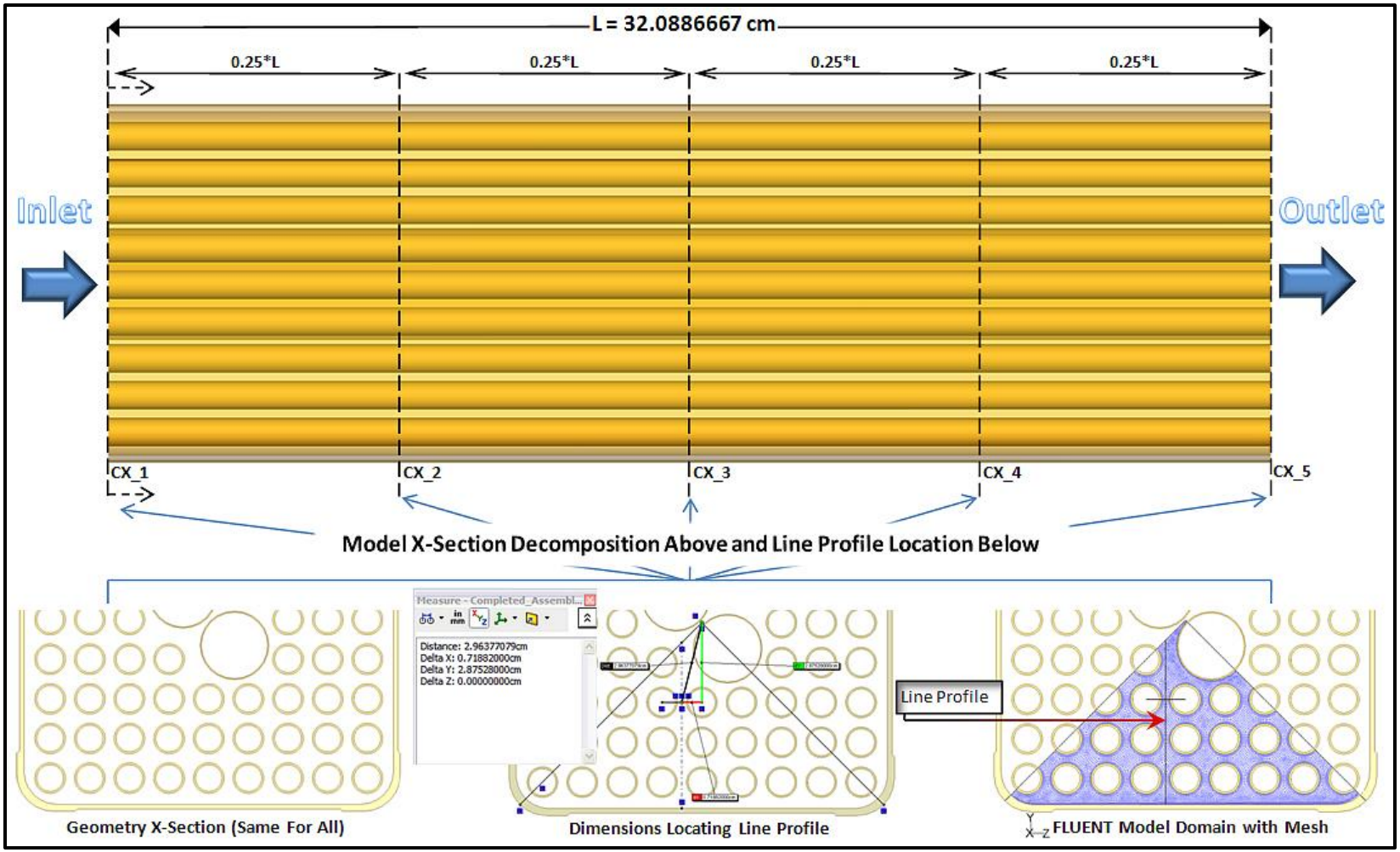


Figure 5.12: Designation of cross-section and line surface profile locations for simulation trial runs and results.

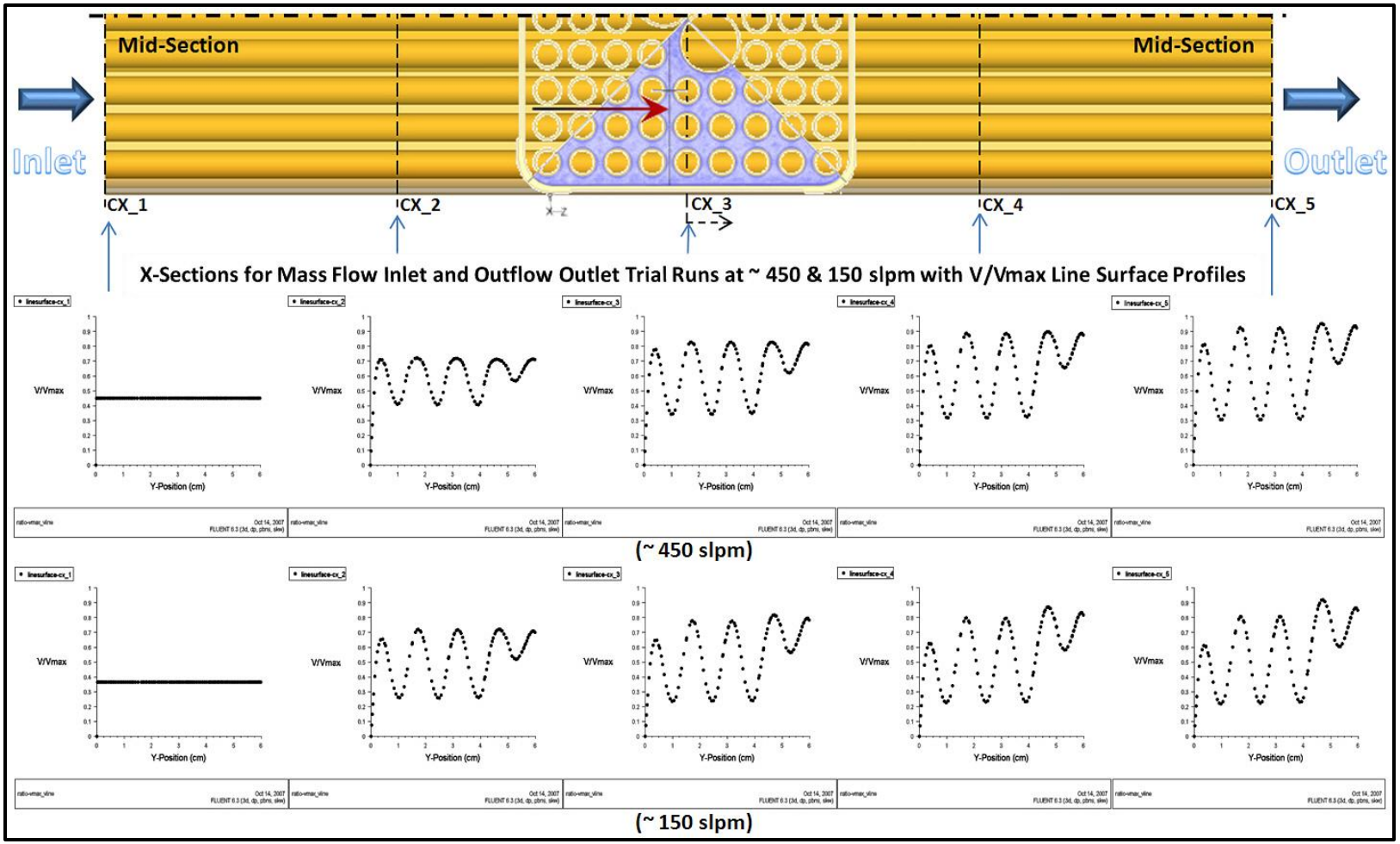


Figure 5.13: Line surface profile results from preliminary trial runs at ~ 450 and 150 slpm for V/Vmax (line surface point velocity/maximum domain velocity) versus y-position (measured from canister wall).

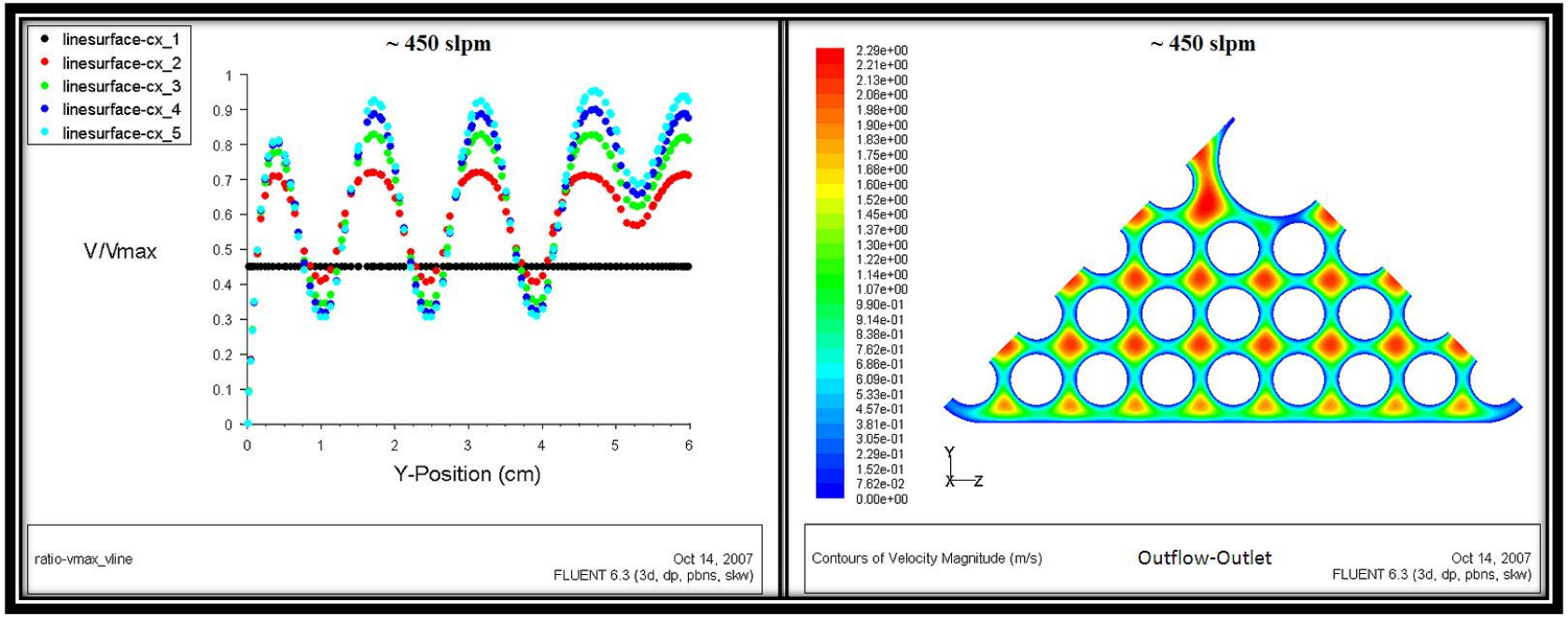


Figure 5.14: Line surface profile results from Figure 5.13 at ~ 450 slpm with accompanying velocity magnitude contour plot at the outflow outlet.

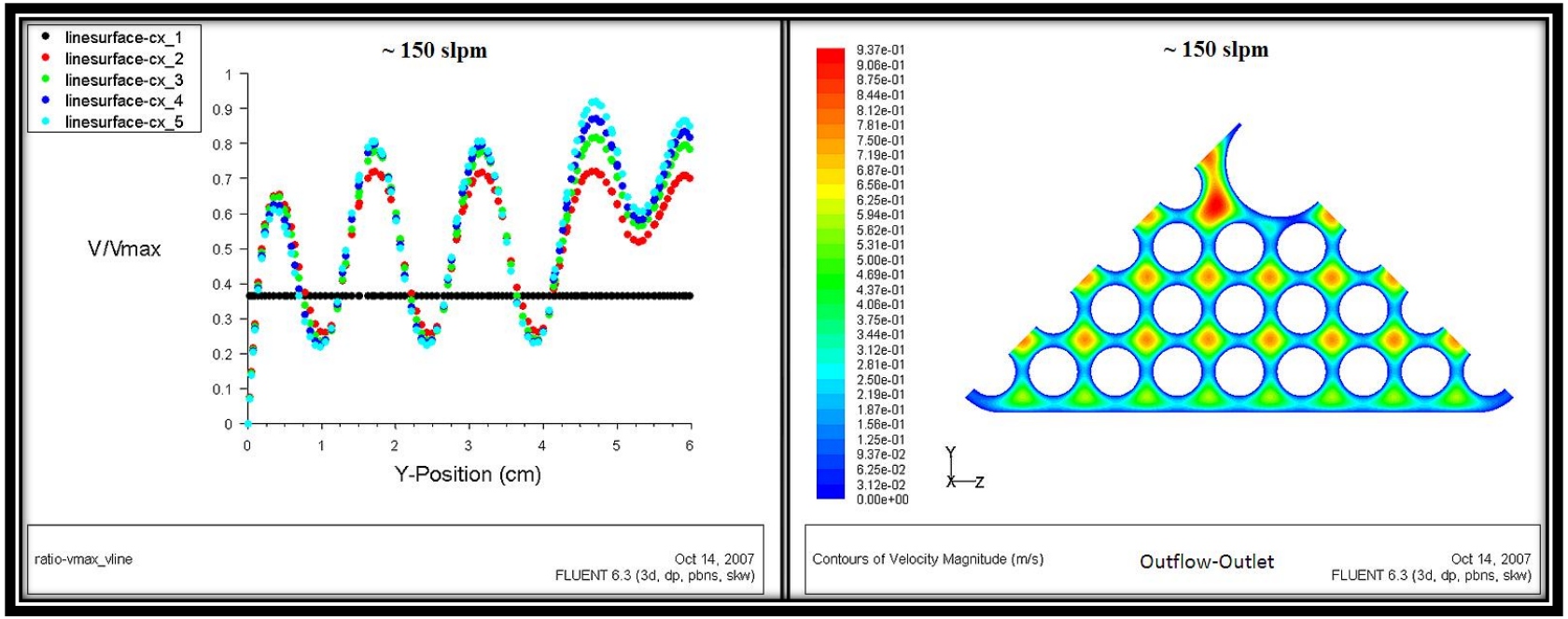


Figure 5.15: Line surface profile results from Figure 5.13 at ~ 150 slpm with accompanying velocity magnitude contour plot at the outflow outlet.

The numerical and graphical solution data presented above is primarily considered on a practical, qualitative level that is most useful as an indicator of the type and form of flow field expected to develop in subsequent refinements of the computational runs based on the solutions sets from simple preliminary test simulations. Inevitably, the overall features of the final flow-field results are subject to variation as higher levels of precision and accuracy are implemented into the numerical model and resolution of the mesh. In summary, a few significant observations afforded by the preliminary case trials performed are outlined below.

Preliminary Case Trial Observations:

- Foremost, a converged solution is obtained for each of the simulation trials with reasonable criteria for convergence imposed. Iteration counts ranged between 300 and 400 before achieving solution convergence.
- Scaled residuals for all equations varied smoothly in each simulation as the calculations progressed toward convergence.
- Average and maximum (y^+) values are well-within acceptable guideline limits for the turbulence model and wall treatment employed.
- The general flow-field features are sensible for the modeled conditions and logically conclude the absence of any critical mesh/model generation errors or inadequacy of imposed boundary conditions to enclose the computational domain. However, an inaccurately applied boundary condition or misapplication of a boundary type is not precluded by this observation alone.
- Furthermore, all graphical contour plots and numerical data trends of velocity magnitude, static pressure, dynamic pressure, and total pressure in Figures 5.6

through 5.15 illustrate realistic fluid flow movement for wall-bounded flow conditions from inlet to outlet.

- Numerical data in Figures 5.6 and 5.7 clearly indicate congruent results between simulations of a specific flow rate with variances in the applied inlet/outlet boundary type(s). This observation is expected if model correctness exists for all case trial variants of inlet/outlet boundary type combination and is indicative of an appropriate domain grid.
- Other expected solution features in Figures 5.8 through 5.11 include relatively uniform static pressures at cross-sections (incompressible, constant area of flow in domain), mirroring trends in velocity and dynamic pressure contours, and diversion of mass flow toward the center of interstitial flow spaces with increasing flow development. This is reinforced by the line surface results in Figures 5.12 through 5.15, which illustrate increasing amplitudes in velocity profile responses in the direction of fluid flow.

These preliminary case trial results impart an increased understanding and impression of the overall solution features expected. The substantial gain in problem clarity facilitates the initial setup and range of mesh sizes selected for evaluation in the grid refinement process. Demonstrating grid independence is a serious consideration in CFD research and establishing the attainment of a grid-independent solution is commonly the ensuing objective.

5.1.2 Grid Independence Study

The most important quality or attribute of any simulated solution set is accuracy, wherein the degree of accuracy is simply a measure of the closeness a numerical solution is to the real physical solution. Obviously, the accuracy is diminished by introducing any form of error into the simulation analysis, which falls under the global term of discretization error in the CFD field of study. In other words, the discretization error is defined as all error associated with a particular numerical method. Therefore, in computational fluid dynamics considerations, the sum of round-off, truncation, and modeling error is equal to the discretization error of the simulation effort as a whole.

Assuming the model is properly setup in terms of its application to the physical flow domain examined, the truncation error is the most significant source of error within the system. Truncation error is directly related to the characterization of an infinite/continuous domain (or its quantities) and governing flow equations by a finite/discrete representation of algebraic equations which are easily solved. This is frequently accomplished by employing a finite number of grid points distributed over the entire flow field through the construction of a finite volume mesh. The primary tool available to a CFD modeler that is capable of directly affecting and improving the accuracy of a simulation is grid refinement (or/along with grid adaption), and is an extremely important step in the process of obtaining an accurate, grid-independent solution; "The matter of grid independence is a serious consideration in CFD…" (Anderson, 1995).

In general, if the flow-field parameter values are quite different for subsequent calculations involving respectable decreases in the grid point increments, then it is highly probable that the solution is a function of the number of grid points utilized, which is an 'untenable' condition (Anderson, 1995). It is imperative at this point, if practical, to

continually increase the number of grid points creating a progressively finer mesh with each refinement until a solution that is no longer responsive (within a set limit of accuracy, tolerance, or variable gradient) to the number of node points in the mesh is established. Consequently, it is permissible to conclude that grid independence is achieved under these conditions although the analytical results may not exactly match the resulting solution. In other words, a grid independence study is simply a numerical uncertainty analysis, which accesses the truncation error of a grid and computational solution in an effort to illustrate the convergence.

Theoretically, the truncation error goes to zero as the total number of nodes approaches infinite value. The theoretical basis behind grid studies, accuracy improvement methods, and fundamentals for judging resultant efficacy is thoroughly discussed in the literature elsewhere (Ball, 2004; Versteeg and Malalasekera, 1995; Anderson, 1995). However, prior to the presentation of any grid study results, a concise introduction and demonstration of notable particulars are provided for reference concerning the general grid independence study conducted as part of this research; to an extent, the information is based on contributions from these sources.

For simplicity, consider the one-dimensional grid example shown in Figure 5.16 below with uniform increments of (Δx) spacing. Focusing on the continuous function *U(x)*, the objective is to construct a differencing scheme using Taylor series expansions around a nodal point specified in the grid, illustrate the sensitivity of solution error to grid refinement, and identify the order of the difference approximation (*i.e.* order of accuracy associated with the differencing scheme). A central difference expression that is second-order accurate is presented in the following derivation for reasons of similarity with solver settings implemented in the majority of FLUENT CFD simulations performed.

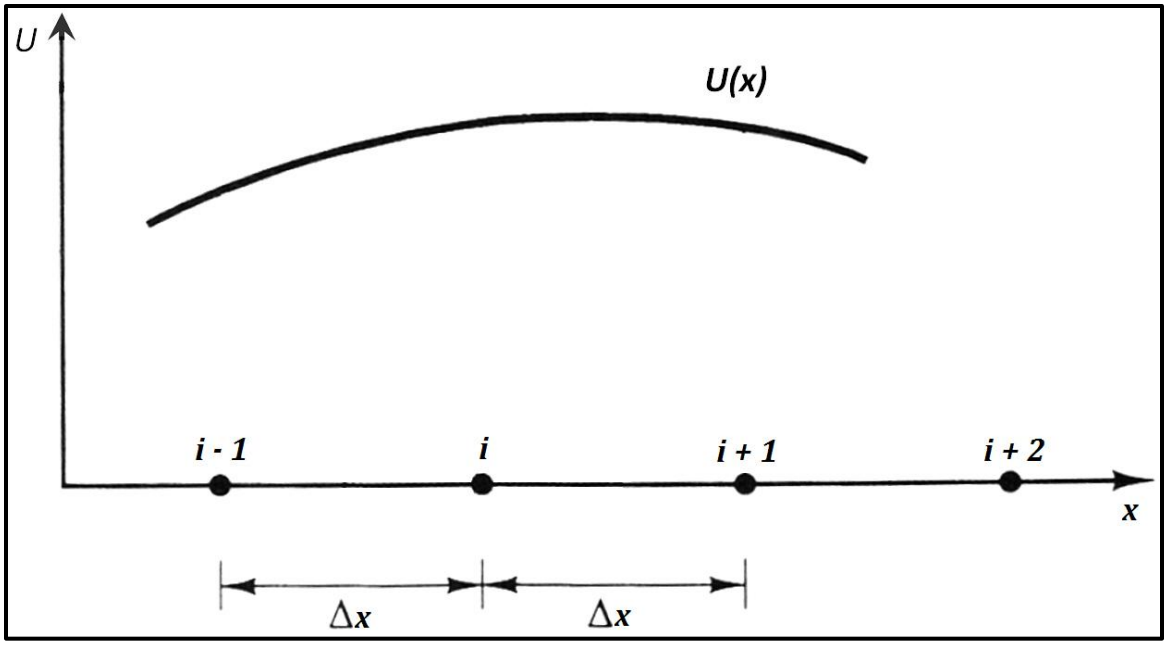


Figure 5.16: One-dimensional grid example with uniform spacing increments between node points; adapted in part from (Versteeg and Malalasekera, 1995).

The Taylor series expansion development of $U(x + \Delta x)$ about the point (i) at (x), for the function *U(x)*, is given by

$$U(x + \Delta x) = U(x) + \left(\frac{\partial U}{\partial x}\right)_x \Delta x + \left(\frac{\partial^2 U}{\partial x^2}\right)_x \frac{(\Delta x)^2}{2} + \left(\frac{\partial^3 U}{\partial x^3}\right)_x \frac{(\Delta x)^3}{6} + \cdots \tag{5.1}$$

where, the following notation is used to define discrete values for *U(x)* and $U(x + \Delta x)$ respectively as

$$U(x) = U_i \tag{5.2}$$

$$U(x + \Delta x) = U_{i+1} \tag{5.3}$$

Using the discrete values from Equations (5.2) and (5.3), Equation (5.1) becomes

$$U_{i+1} = U_i + \left(\frac{\partial U}{\partial x}\right)_i \Delta x + \left(\frac{\partial^2 U}{\partial x^2}\right)_i \frac{(\Delta x)^2}{2} + \left(\frac{\partial^3 U}{\partial x^3}\right)_i \frac{(\Delta x)^3}{6} + \cdots \tag{5.4}$$

Similarly, the Taylor series expansion development of $U(x - \Delta x)$ about the point (i) at (x), is given by

$$U(x - \Delta x) = U(x) - \left(\frac{\partial U}{\partial x}\right)_x \Delta x + \left(\frac{\partial^2 U}{\partial x^2}\right)_x \frac{(\Delta x)^2}{2} - \left(\frac{\partial^3 U}{\partial x^3}\right)_x \frac{(\Delta x)^3}{6} + \cdots \tag{5.5}$$

where, the following notation is again used to define discrete values for *U(x)* and $U(x - \Delta x)$ respectively as

$$U(x) = U_i \tag{5.6}$$

$$U(x - \Delta x) = U_{i-1} \tag{5.7}$$

Using the discrete values from Equations (5.6) and (5.7), Equation (5.5) becomes

$$U_{i-1} = U_i - \left(\frac{\partial U}{\partial x}\right)_i \Delta x + \left(\frac{\partial^2 U}{\partial x^2}\right)_i \frac{(\Delta x)^2}{2} - \left(\frac{\partial^3 U}{\partial x^3}\right)_i \frac{(\Delta x)^3}{6} + \cdots \tag{5.8}$$

Finally, with values defined for the grid points at (Δx) intervals to the left and right of (i), a central difference formula for the gradient evaluation at midpoint (i) is attainable. This is realized by subtracting Equation (5.8) from Equation (5.4) to give

$$U_{i+1} - U_{i-1} = 2\left(\frac{\partial U}{\partial x}\right)_i \Delta x + 2\left(\frac{\partial^3 U}{\partial x^3}\right)_i \frac{(\Delta x)^3}{6} + \cdots \tag{5.9}$$

Subsequently, the resultant expression [Equation (5.9)] is solved for the partial derivative at (*i*), yielding

$$\left(\frac{\partial U}{\partial x}\right)_i = \frac{U_{i+1} - U_{i-1}}{2\Delta x} - \left(\frac{\partial^3 U}{\partial x^3}\right)_i \frac{(\Delta x)^2}{6} - \cdots \tag{5.10}$$

An alternate form of Equation (5.10) is also stated as

$$\left(\frac{\partial U}{\partial x}\right)_i = \frac{U_{i+1} - U_{i-1}}{2\Delta x} + truncated\ terms \tag{5.11}$$

Hence, in Equation (5.11), the first term on the right-hand side of the equality is a central difference representation of the partial derivative and the remaining (truncated) terms constitute the truncation error. In other words, if the partial derivative is approximated with the above algebraic difference quotient denoted

$$\left(\frac{\partial U}{\partial x}\right)_i \approx \frac{U_{i+1} - U_{i-1}}{2\Delta x} \tag{5.12}$$

at node point (*i*), then the error specified by the truncated terms in Equation (5.10) is a clear indication of the accuracy associated with this approximation. As implied, the truncation error is directly correlated with the neglect of higher-order truncated terms in

the difference expression and is reduced by incorporating increasingly finer mesh resolutions or decreased increments of (Δx) between successive grid points.

The rate at which the error tends to zero with further mesh refinement is governed by the power of (Δx) for the lowest-order term in the truncation error, which is also defined as the order of the difference approximation. Employing a mathematical notation symbol of general form $O(\Delta x)^n$, with (n) specified as the power of (Δx) for the lowest-order term in the truncation error, a formal representation of Equation (5.10) is rewritten in a more precise notation than Equation (5.12) as follows:

$$\left(\frac{\partial U}{\partial x}\right)_i = \frac{U_{i+1} - U_{i-1}}{2\Delta x} + O(\Delta x)^2 \tag{5.13}$$

Therefore, with the lowest-order term in the truncation error containing $(\Delta x)^2$ in Equation (5.13), the central difference is second-order accurate and the final form of the difference quotient expressed by Equation (5.13) for the partial derivative is labeled as a 'second-order central difference' (Anderson, 1995).

This finite-difference derivation is sufficiently similar to the calculations performed by the FLUENT solver to impart an adequate understanding of numerical characteristics necessary for the subsequent discussions of grid independence. Again, the grid study allows a CFD modeler to easily access, illustrate, and judge the accuracy of a numerical scheme utilized in simulations. Accordingly, the uncertainty analysis retrieves the truncation error with the primary objective of exhibiting the following two computational features: (1) solution convergence, indicated by a condition in which the truncation error tends to zero as an infinite number of grid points is approached (*i.e.* $\Delta x \rightarrow 0$) and (2) order of accuracy associated with the difference expression utilized. This

confirms or negates agreement between the simulation results and the mathematical relations/numerical scheme details inherent to a particular option method selected in the solver setup.

In order to draw these conclusions, the identification of a clearly defined, measurable quantity of error for a parameter with significant flow-field dependence is initially required for comparative analysis. Typically, there are two option types available for consideration: (1) selection of a local quantity (*i.e.* temperature) in which a L2 or root mean squared (RMS) error is defined and (2) selection of an integral quantity (*i.e.* surface heat flux) in which an absolute and relative error are defined. The error quantity measurement chosen for the grid independence study conducted in this research is the area-weighted average (AWA), surface integral, static pressure drop across the specified flow domain segment, which is an obvious integral quantity. This decision is based on the reasoning discussed below.

Early in the grid study analysis, the use of a local measurement quantity to calculate the solution error at each grid point in the mesh proved extremely time consuming with such large, three-dimensional grids and required an extensive amount of hard disk space for data storage. Furthermore, an exact local quantity (*i.e.* velocity) in which to compare with resultant simulations does not exist at the same node location within the computational domain for each mesh utilized in the grid study analysis. This circumstance is applicable on a number of different levels toward potential definition of the local error. Foremost is the preferred method in which experimental measurements at each location are used as the exact values for calculating differences with the local quantity results from each grid refinement examined. However, this approach is impractical since the resolution of local measurement quantities from SNL experiments is insufficient for comparison even with the coarsest computational mesh. As a result,

despite the possibility of employing an interpolation scheme for acquiring local values at node locations where exact experimental measurement data is absent, the use of this method for each grid size analyzed is beyond consideration.

Similarly, opting to use node point quantities from the finest mesh solution as the exact values for comparison with local values from grids of varying resolution fails to change the situation. In this case, the mismatched locations of local quantity values compared from each mesh domain persist and any inconsistencies of incorporated interpolation schemes undoubtedly introduce additional error into the analysis. The only possible alternative in which this error is circumvented requires a completely structured set of grids for representing the entire domain volume and that each subsequent grid adaption (*i.e.* change in Δx, Δy, and Δz between mesh nodes) is an even fraction of all other nodal increments (in the same direction) for each grid used in the study.

On the other hand, an interpolation scheme in certain cases introduces minimal inconsistencies into the grid independence study if local quantities from three-dimensional models are pertinent to the error measurement. Unfortunately, this necessitates writing a user-defined function (UDF) in the C-programming language that is compiled into the FLUENT solver code in order to dump local parameter values at specified locations into an external file for determining the error value at each point. In addition, further interpolation is required by the FLUENT solver at locations not coinciding with an existing node or cell face where the sought variable is stored. For models involving significantly large mesh sizes, this inevitably causes an appreciable increase in the time required to post-process the data as well as the expenditure of disk space for file storage.

Due to inherent complexity of the three-dimensional fuel assembly domain, large cell volume counts for even the moderate grid resolutions, and a need to limit preliminary

resources allotted for postprocessing of simulation data, the use of an integral quantity in the determination of computational error is deemed the most adequate and viable option. As previously stated, the integral quantities taken are AWA static pressures over all cell faces at the inlet/outlet planes of the modeled domain, which are an appropriate selection for the comparison of pressure drop solutions since the flow is assumed incompressible (final steady state) and the fuel assembly segment evaluated in the grid study has a constant cross-sectional area. Thus, error quantities are simply based on comparisons between differential static pressure values across the inlet and outlet planes of each mesh examined. These pressure loss solutions are also of utmost importance for comparative analysis with SNL experimental measurements. Accordingly, the absolute and relative errors for the grid independence study are defined as follows:

$$Error_{absolute} = \varepsilon_a = \left| P_{calc}^{loss} - P_{exact}^{loss} \right| \tag{5.14}$$

$$Error_{relative} = \varepsilon_r = \frac{\varepsilon_a}{P_{exact}^{loss}} = \frac{\left| P_{calc}^{loss} - P_{exact}^{loss} \right|}{P_{exact}^{loss}} \tag{5.15}$$

Upon establishing a suitable measurement of error, the considerations for conducting a grid independence study and presenting the results are focused on targeting the order of the finite difference formula utilized in computing the gradient terms of interest. In most CFD applications, first-order accuracy is insufficient and a more accurate difference representation is necessary (Anderson, 1995), especially if the flow environment modeled is expected to include large gradients for characteristic parameters or complicated multipath/mode transport of flow-field variables that are interdependent (Versteeg and Malalasekera, 1995). The anticipated complexity of developed airflows

traversing the fuel assembly domain warrants the use of second-order accurate evaluations for all cell face quantities.

Therefore, the second-order discretization option for the pressure interpolation scheme is selected in the solver controls as well as the second-order upwind discretization for momentum, turbulent kinetic energy, and specific dissipation rate equations. Referring to the second-order central difference example outlined above in which Equation (5.13) is derived, the decision to employ these second-order schemes ultimately implies the specified power (n) of (Δx), for the lowest-order term in the truncation error, has a value of two. In other words, regardless of the variable designated in the partial derivative (PD) expression or the type of difference quotient (must be second order) used in these governing equations, if the mesh is sufficiently resolved and boundary conditions correctly applied, then a valid form of the gradient equation is

$$PD_{(exact\ at\ i)} = Difference_{quotient} + O(\Delta x)^2 \tag{5.16}$$

The task of illustrating both graphically and mathematically the process of exploiting a grid independence study to provide evidence that numerical results obtained from the difference equations are within acceptable limits (at least the truncation error) of those acquired from a closed-form analytical solution of the originally approximated partial derivative expression is relatively straightforward. Again, this is most-efficiently accomplished by accessing the truncation error to demonstrate solution convergence (*i.e.* as $\Delta x \rightarrow 0$, $Error_{truncation} \rightarrow 0$) and order of accuracy for the difference representation. Therefore, by successively resolving the flow field on grids of increasing resolution and defining the error as the difference between AWA pressure loss values obtained for each mesh size (calculated values) and the benchmark value from the highest resolution mesh

(exact value), a converging slope becomes evident on the graph consisting of relative error versus total number of grid points. Obviously, this convergence characteristic is stipulated on the implementation of an appropriate numerical model for the actual physics at work within the flow domain.

Presenting the order of accuracy from the same graphical plot is easily achieved in a few mathematical steps by re-emphasizing the defined error in general terms as follows:

$$\varepsilon = PD_{(exact\ at\ i)} - Difference_{quotient} = \mathrm{O}(\Delta x)^2 \tag{5.17}$$

or, using the initial example described above, this becomes

$$\varepsilon = \left(\frac{\partial U}{\partial x}\right)_i - \frac{U_{i+1} - U_{i-1}}{2\Delta x} = \mathrm{O}(\Delta x)^2 \tag{5.18}$$

Subsequently, the expected characteristic slope is revealed by taking the natural logarithm of the error statement [same for Equations (5.17) and (5.18)], expressed as

$$\ln \varepsilon = \ln[\mathrm{O}(\Delta x)^2] \tag{5.19}$$

where, for a uniform (or nearly uniform) grid of (Δx) spacing and model length in the specified direction (L_{m_x}), the number of mesh intervals (N) is given by

$$N = \frac{L_{m\_x}}{\Delta x} \tag{5.20}$$

Furthermore, in contrasting the errors associated with models of identical solver control specifications and domain geometry, such that mesh interval spacing (Δx) is the only parameter directly varied between models, Equations (5.19) and (5.20) are adequately replaced by approximate relationships that simplify the comparison of error quantities and are respectively stated as follows:

$$\ln \varepsilon \sim \ln(\Delta x)^2 \tag{5.21}$$

$$N \sim \frac{1}{\Delta x} \tag{5.22}$$

Combining these approximate relations with known mathematical properties of natural logarithms, yields

$$\ln \varepsilon \sim \ln\left(\frac{1}{N}\right)^2 = -2\ln(N) \tag{5.23}$$

As a result, the ideal characteristic slope for a best-fit line to the plotted data of relative error versus total number of grid points/cells (on logarithmic-scaled axes) for successive mesh refinements and a second-order accurate differencing scheme is

$$\frac{\ln(\varepsilon_r)}{\ln(N)} = \frac{\log(\varepsilon_r)}{\log(N)} = \frac{\varepsilon_{r(\log\_scale)}}{N_{(\log\_scale)}} = -2 \tag{5.24}$$

Consequently, a second-order accurate representation of the partial derivative using an algebraic system of difference quotients in which the dependent parameter of interest is sufficiently resolved throughout the flow field, exhibits two essential attributes concluded

from a grid independence study: (1) solution convergence and (2) a second-order accurate characteristic slope (≈ -2) for the graphical trend of resulting error versus grid size data (log-scaled axes). Obviously, stipulations for observing these attributes also consist of a properly constructed model for the type of flow field anticipated with suitable boundary conditions defined at the computational domain extents.

Other considerations concerning the methodology and equations detailed in the above example include two points of noteworthy emphasis involving the round-off error and exactness of derivations toward gradient scheme calculations in the FLUENT CFD solver. For the round-off error, it simply suffices through brief mention that these error quantities are not directly accounted for in the example despite inherent presence in all simulation results (indeed, any computer-generated solution) as this contribution of error is reasonably assumed negligible with the use of double precision in all simulations. Moreover, the example derivations and demonstration of numerical significance imparted by a grid independence study are intentionally based on an overtly simplified, finite-difference analysis involving a one-dimensional, uniformly spaced mesh.

Clearly, the number and complexity of difference expressions for the evaluation of gradients specified within the FLUENT solver for three-dimensional models are significantly increased and differences in the derived equation forms between finite-difference and finite-volume approaches are inevitable. However, despite variations between the illustrated example and exact formulations of the CFD solver, presenting a more detailed/complex example of identical derivation for gradient expressions is unnecessary. The associated thought process and concept of differencing scheme application are fundamentally analogous in both cases and ultimately lead to parallel conclusions concerning grid independence assessments and order of accuracy identification.

In summary, cumulative consideration for the above examples, equation derivations, and numerical characteristics of proper grid independence studies provides a sufficient foundation for the general understanding and interpretation of the grid study results presented. This research effort includes two separate refinement studies conducted near the high- and low-value extents of the designated flow rate range (approximately 450 and 150 slpm, respectively) in order to provide reasonable justification for any conclusions drawn from the solution set of both studies (*i.e.* adequate mesh resolution) as sufficiently applicable for the entire range of airflow rates spanned. Using the relative error definition given by Equation (5.15), a description of the pertinent simulation details, boundary conditions employed, and tabulated numerical results for both grid study assessments are provided in Tables 5.1 and 5.2 below with graphical presentations of these results in Figures 5.17 and 5.18, which follow. Small cross-sectional mesh images located at a bottom-corner quadrant of the flow domain are also provided in Figure 5.19 below as a visual reference to the mesh size/resolution implemented in each of the grid refinement trials.

Table 5.1: Reference model values and simulation results for grid independence study (1 of 2).

[SETUP CONDITIONS AND RESULTS FOR GRID INDEPENDENCE STUDY]

Reference Values

	Value:
Air Density (kg/m^3)	0.98
Air Viscosity (kg/s-m) $_{dynamic}$	1.85E-05
Flow Area (m^2) $_{x\text{-}section}$	0.00968829
Wetted Perimeter (m) $_{x\text{-}section}$	3.254596
Hydraulic Diameter (m)	0.011907
Turbulence Model	k-w standard $_{transitional}$
Total Model Length (m) $_{domain}$	0.32088667

Boundary Conditions - Mass Flow Inlet / Pressure Outlet

FLOW RATE [~ 450 (SLPM)]	
Variable:	Value:
Domain Flow (standard liters per minute)	454
Air Mass Flow Rate (kg/s) $_{inlet}$	0.0098
Turbulence Intensity (%) $_{inlet}$	7.1190
Gauge Pressure (Pa) $_{outlet}$	0
Turbulence Intensity (%) $_{outlet}$	7.1190

FLOW RATE [~ 150 (SLPM)]	
Variable:	Value:
Domain Flow (standard liters per minute)	151
Air Mass Flow Rate (kg/s) $_{inlet}$	0.003267
Turbulence Intensity (%) $_{inlet}$	0.1
Gauge Pressure (Pa) $_{outlet}$	0
Turbulence Intensity (%) $_{outlet}$	0.01

Table 5.2: Reference model values and simulation results for grid independence study (2 of 2).

Results / Comparisons

Note: Appropriate aspect ratios for volume cells are maintained in the flow direction, while the number of face cells on normal boundary plane are varied.

FLOW RATE [~ 450 (SLPM)]

		Grid Study Plot Data			Additional Data - Reference Only	
Trial:	**Face$_{x\text{-section}}$ Cell Count:**	**Total Node Count:**	**Pressure Loss (Pa):**	**Relative Error ():**	**Max Velocity (m/s) $_{outlet}$:**	**Relative Error ():**
1	5317	1439130	2.607622	0.02570778	2.264266	0.01032615
2	6461	1629654	2.622348	0.02020578	2.275908	0.00523775
3	7391	1752144	2.633120	0.01618075	2.282489	0.00236148
4	7935	2754216	2.639734	0.01370966	2.284187	0.00161922
5	8667	3030882	2.656901	0.00729566	2.279060	0.00386002
6	10488	3615423	2.668659	0.00290234	2.285606	0.00099887
Benchmark	12959	7016004	2.676427	0	2.287891	0

FLOW RATE [~ 150 (SLPM)]

		Grid Study Plot Data			Additional Data - Reference Only	
Trial:	**Face$_{x\text{-section}}$ Cell Count:**	**Total Node Count:**	**Pressure Loss (Pa):**	**Relative Error ():**	**Max Velocity (m/s) $_{outlet}$:**	**Relative Error ():**
1	5317	1439130	0.726320	0.03078198	0.922752	0.01201800
2	6461	1629654	0.732657	0.02232648	0.922264	0.01254046
3	7391	1752144	0.736524	0.01716615	0.927121	0.00733966
4	7935	2754216	0.738458	0.01458579	0.930230	0.00401148
5	8667	3030882	0.744219	0.00689839	0.932053	0.00205877
6	10488	3615423	0.747655	0.00231273	0.933740	0.00025291
Benchmark	12959	7016004	0.749388	0	0.933976	0

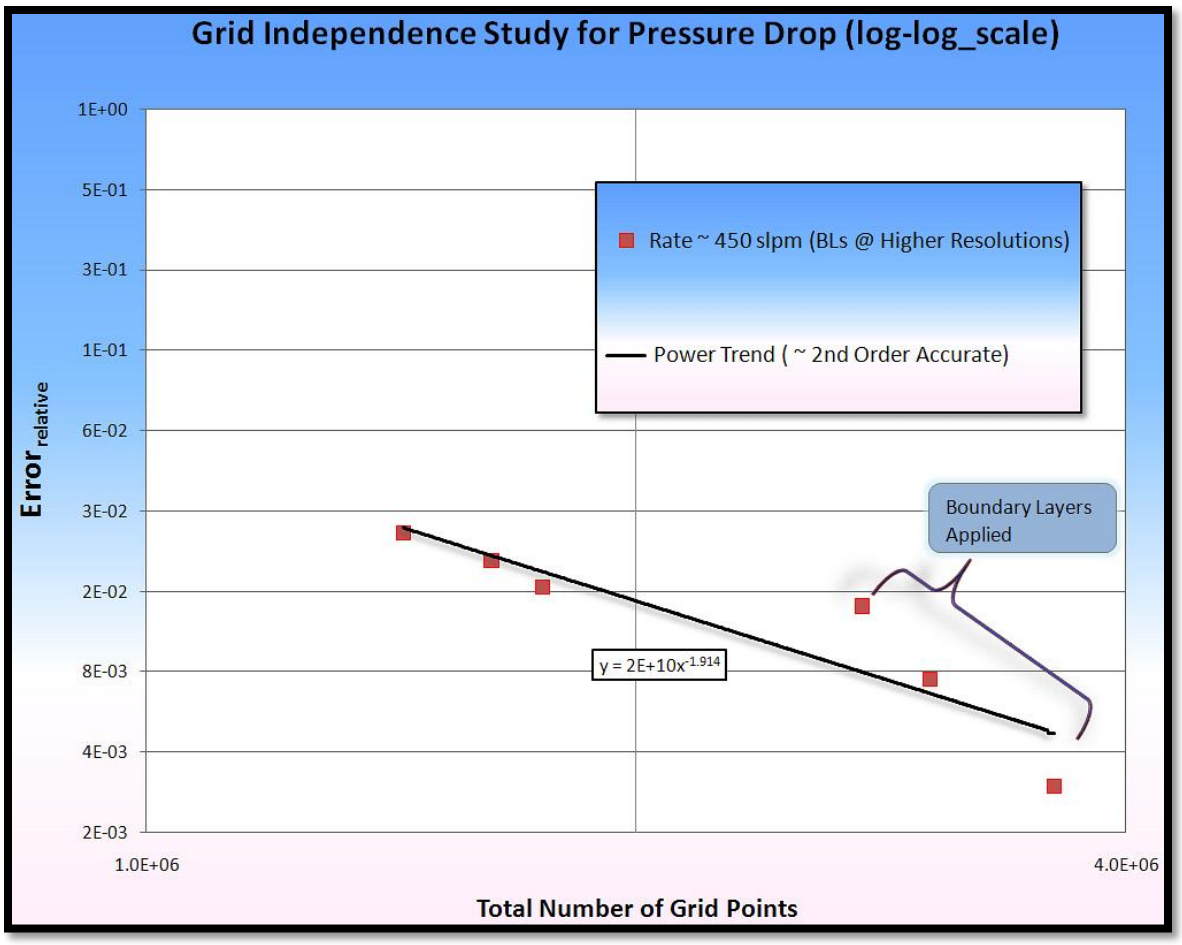


Figure 5.17: Simulation results for relative error versus total number of grid points at a flow rate of ~ 450 slpm.

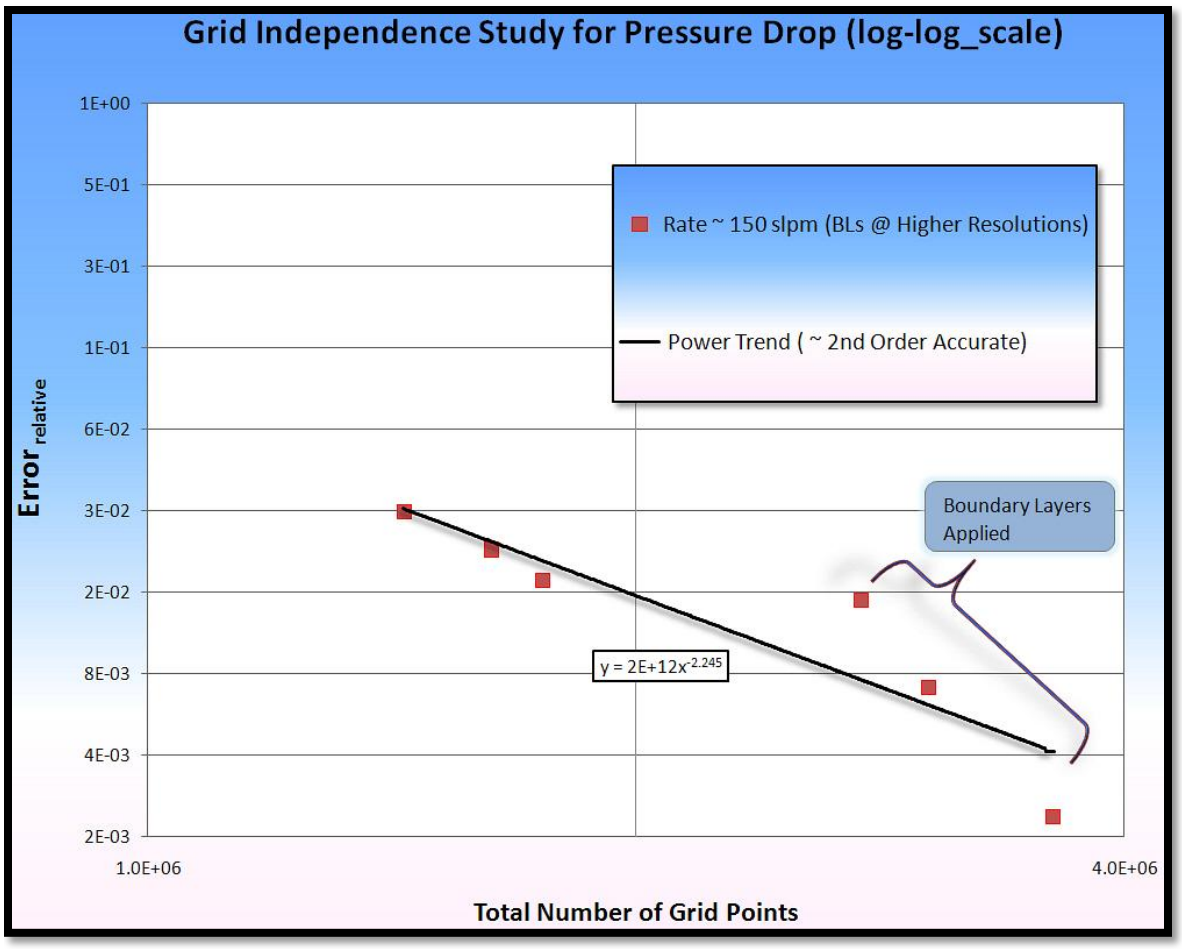


Figure 5.18: Simulation results for relative error versus total number of grid points at a flow rate of ~ 150 slpm.

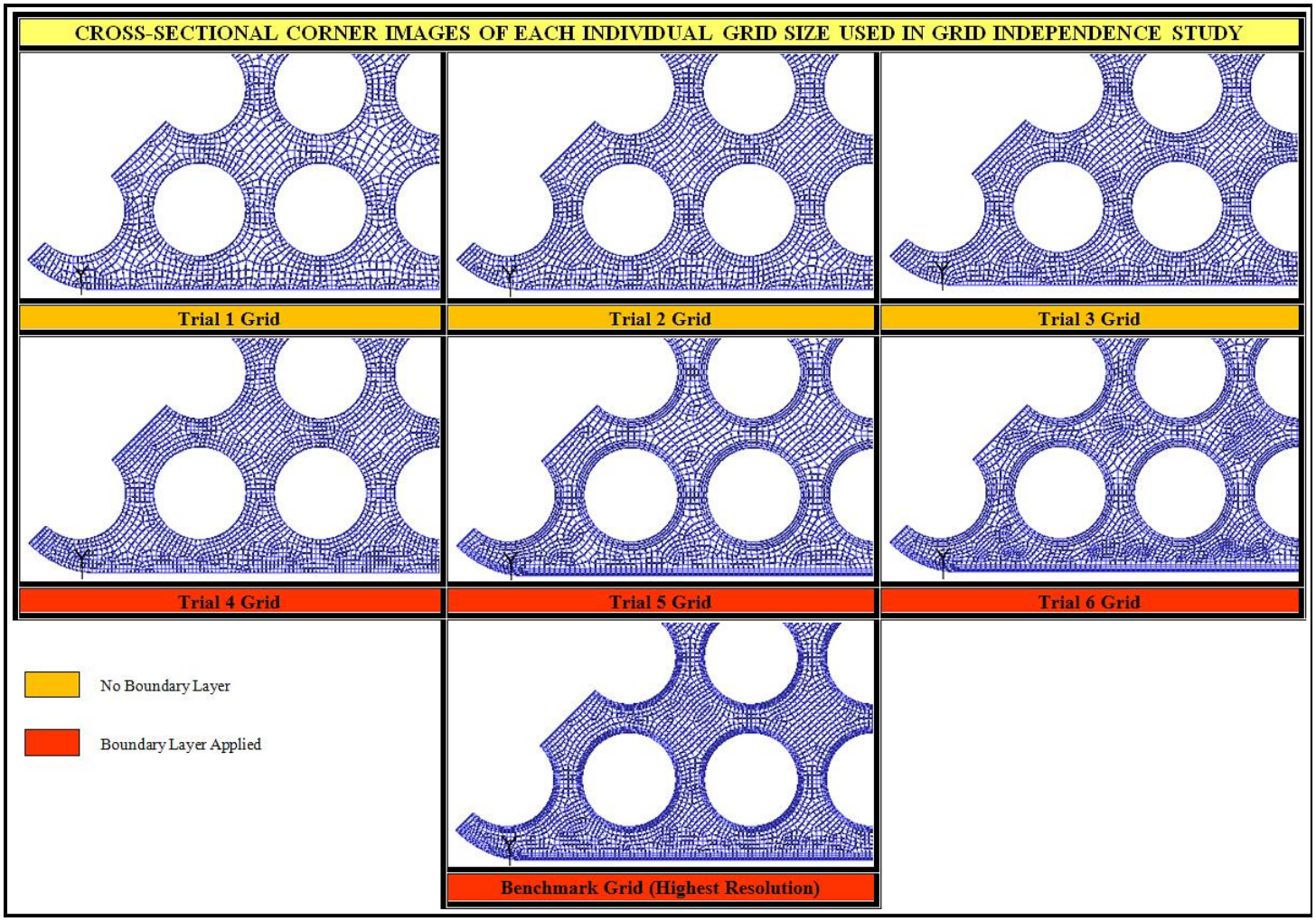


Figure 5.19: Reference mesh size images at bottom-corner quadrant of flow domain for each grid study trial.

The simulation results from the grid independence study at airflow rates of approximately 450 and 150 slpm, displayed in Tables 5.1 and 5.2 as well as Figures 5.17 and 5.18, offer several different insights into the predicted system response for pressure loss with respect to the total number and distribution of cells within a specified mesh. This is especially evident in the analysis of near-wall flow fields as these two grid qualities are adjusted and further affected by the implementation of any boundary layer cells at such locations. However, as far as the present grid study is concerned, focus is directed upon the evaluation of aforementioned objectives stated as the primary impetus for conducting this assessment; these include substantiation of solution convergence with increased grid refinement and second-order accurate approximation of flow-field gradients.

The former, characterized by a difference expression approaching an exact representation of the original partial derivative (gradient) as nodal spacing tends to zero (*i.e.* $\Delta x \rightarrow 0$), is exemplified through the continuous reduction of relative error as the mesh size/resolution is successively increased for each grid refinement trial. This feature of the numerical model is clearly demonstrated by the tabulated solution values (see Tables 5.1 and 5.2) as well as graphical results (see Figures 5.17 and 5.18) from the grid independence study shown above and concludes adequate solution convergence at both airflow rates examined (*i.e.* as $N \rightarrow \infty$, $\text{Error}_{\text{truncation}} \rightarrow 0$). Furthermore, considering the data points for grids with and without boundary layers separately in Figures 5.17 and 5.18 reveals a higher rate of convergence for the grids containing a boundary layer cell structure compared to the more-uniform meshing scheme present in the other grids analyzed. Although this benefit comes at the expense of larger grid sizes to reach comparable solution differences with the benchmark result, the increased convergence rate is highly desirable and outweighs the additional resource costs; especially with

considerations for near-wall mesh resolution and its contributions toward greater accuracy in resolved pressure losses due to viscous effects at bounding walls. Therefore, the use of boundary layers near fuel rod surfaces and outer canister walls is emphasized in remaining mesh model constructions.

The latter objective, establishing second-order accuracy, is accomplished by inspecting the slope of the linear trend in Figures 5.17 and 5.18 for relative error versus number of grid points across the mesh refinement levels. Despite the relatively poor fit to the data resulting from varying convergence rates between points associated with and without boundary layer inclusion, the fitted line is primarily intended to serve as a simple reference upon which a general trend in the data set is discerned. Again, as expressed in Equation (5.24) above, a second-order accurate scheme is expected to exhibit a slope of nearly negative two or less (*i.e.* slope < -2) for the data points plotted on log-scaled axes. For both flow rates assessed, the observed slope of a linear trend to the data in Figures 5.17 and 5.18 is approximately negative two for grids without boundary layers and even less (more pronounced negative slope) for the grids with boundary layers incorporated.

If not already apparent, the slope for a line of best fit expressing the linear relationship between data points transformed on log-scaled axes is provided by the exponent of the best-fitting power curve to the data; this is the resulting equation illustrated in Figures 5.17 and 5.18 using the power function trend. A straightforward explanation of this correlation is demonstrated through brief consideration of the power function in which a general form of the equation is written as follows:

$$y = bx^m \tag{5.25}$$

where, (y) and (x) are termed the y- and x-axis values accordingly with parameters (b) and (m) defined as constants. Subsequently, by taking the logarithm of both sides, Equation (5.25) becomes

$$\log(y) = \log(bx^m) \tag{5.26}$$

and rewriting in a modified form with slight rearrangement, yields

$$\log(y) = \log(b) + \log(x^m) \tag{5.27}$$

$$\log(y) = \log(b) + m\log(x) \tag{5.28}$$

Hence, the form of Equation (5.28) illustrates the slope (m) equivalence of the power function for data points transformed on log-scaled axes to the linear relationship stated in the general equation of a line on uniform-scaled axes as

$$m\log(x) + \log(b)\left\{power_{log-scale}\right\} \rightarrow m(x) + (b)\left\{linear_{uniform-scale}\right\} \tag{5.29}$$

Therefore, the results shown in Figures 5.17 and 5.18 afford reasonable justification of an implemented second-order (or higher) accurate discretization scheme and substantiation of solution convergence with increased grid refinement. Furthermore, the mesh resolution characterized by the trial 5 grid is concluded as possessing the most appropriate balance between solution accuracy and computational resource cost with results differing by less than 1% with those obtained by the benchmark grid. Consequently, this grid representation of the modeled flow domain is utilized throughout

the remaining simulations conducted as part of the computational experiment studies previously outlined in Figure 5.2 above.

On a final note, there is notable reason for additionally emphasizing the importance of the mesh skewness ascribed to volume cell elements comprising a computational domain, which also serves as a significant contributor toward the total truncation error and is equally detrimental to the accuracy of any simulation solution. However, a detailed discussion with regard to grid skewness is not addressed due to commonplace of the subject in numerous mathematical applications far outreaching the CFD field of study; such details are deflected to explanations provided in the GAMBIT and FLUENT user's guides accompanying the software (FLUENT Inc., 2006). It simply suffices at this juncture to mention that all meshes utilized throughout CFD research presented in this dissertation are within recommendations for exceptional grid quality structure as it pertains to 3D hexahedral cell elements, primarily assessed in accordance with the equiangle skew (EAS) quality metric and limit (*i.e.* $EAS_{max} < 0.5$) stated in the GAMBIT user's guide (FLUENT Inc., 2006). A snapshot example of this mesh quality check in the GAMBIT software applied to the trial 5 grid is included for reference in Figure 5.20 below.

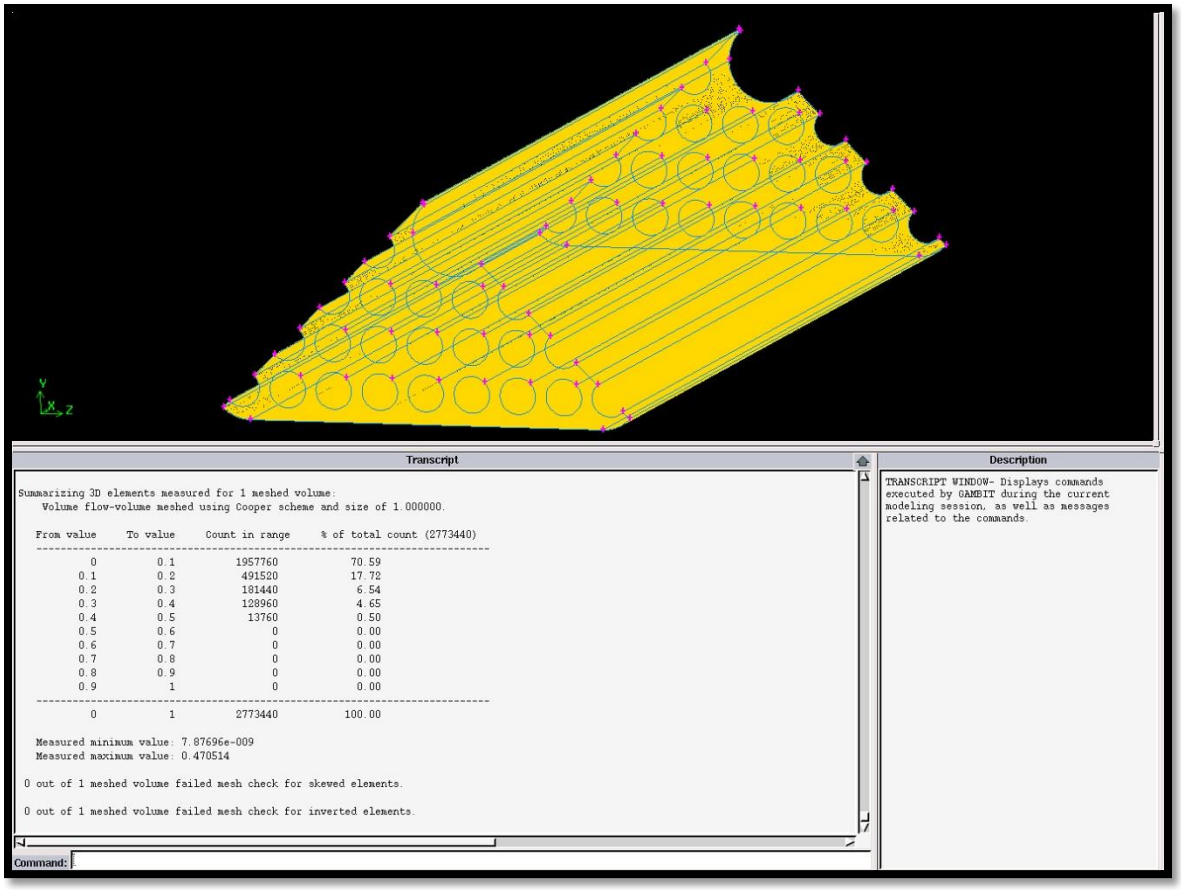


Figure 5.20: Example mesh quality check in GAMBIT for the trial 5 grid with an equiangle skew metric employed.

The maximum EAS value for the hexahedral elements of the trial 5 grid is approximately 0.47 with an average value of less than 0.1 as shown in Figure 5.20 from the resulting mesh check summary. Repeated quality checks (as demonstrated in this example) are performed throughout the mesh construction process in order to ensure that high standards for cell skewness values are upheld in every grid fabricated, while further maintaining reasonable specifications for other prominent mesh quality-type metrics such as the size change and aspect ratio of volume cell elements.

5.1.3 Periodicity and Symmetry Check

The preliminary runs and grid independence study as well as the periodicity investigation and symmetry check are all critical elements in the effort to establish a well-founded base upon which subsequent larger-scale simulation models are progressively built. However, these computational examinations are important for completely different reasons altogether when considering how each contributes to the overall culmination of research objectives. Specifically, the preliminary trial runs and grid independence study are pivotal for imparting evidence in support of an acceptably accurate base model and the general knowledge from which detailed final results stem in subsequent analyses. On the other hand, the symmetry and periodicity investigations serve as demonstrations of two distinctly powerful modeling tools available within the FLUENT CFD software, which are capable of significantly extending the maximum permissible domain size per computational resource limit and adding the ability to predict fully-developed flow-field solutions under varying source conditions with minimal boundary specifications.

Accordingly, with the preliminary trial results and grid independence study discussed in detail, focus is shifted towards a more in-depth set of simulations and flow development studies with regard to symmetry and periodic boundary condition application in fuel assembly models. Directing initial attention on the subject of symmetry boundary conditions, the purpose for conducting a series of symmetry verification calculations is simply to ensure proper implementation within the constructed model and use in representing the anticipated flow-field development. Symmetry boundary conditions are permissible in flow domains where the physical geometry as well as the developed thermal fluid response both exhibit (or are expected to exhibit) a mirrored pattern at the applied boundary plane. There are no input requirements for the specification of this boundary type; however, a great deal of consideration is necessary in

order to assure the selection of an appropriate location for application of this boundary condition within the modeled domain.

The numerical solver calculations are predicated on the assumption of a zero 'flux of all quantities across a symmetry boundary' and 'shear stress in viscous flows' requiring no convective flux; meaning the normal velocity component at the boundary is zero and treatment of the physical boundary plane is analogous to a 'slip' wall for viscous related flows (FLUENT Inc., 2006). As previously discussed, conditions appropriate for this boundary type are expected to exist within the modeled fuel bundle domain across designated 90° intervals from any plane located at 45° in either direction, measured from the horizontal z-axis in a 2D cross-section view of the fuel assembly. The implementation of a symmetry boundary at these planes allows a 75% reduction in the model size over the actual physical extents of the fuel assembly structure, alleviating an otherwise enormous strain on computational resources and scheduling commitment. However, this initial reduction in the flow domain size is counteracted to a minute degree by increased grid resolution requirements near these boundaries to ensure adequate resolution of the flow field on either side of the mirrored plane, but at a minimal cost in comparison to the overall benefit afforded.

Therefore, to establish a reasonable level of confidence in the assertion of sufficient node/cell clustering at these mirrored symmetry boundaries, the solutions from simulations employing both symmetry imposed and complete mesh model representations of the bundle flow domain at a higher-end airflow rate of approximately 450 slpm are compared. Furthermore, implications associated with the misapplication, ill-posed location, or overlooked incongruence of the flow field concerning the symmetry boundary conditions are exemplified by any dissimilarities in the simulation results. Reference values for these simulations and a comparison of resulting parameter solutions

obtained from both model implementations are presented in Table 5.3 below with several depictions of each mesh model in Figures 5.21 and 5.22, which follow.

Table 5.3: Reference values and comparison of numerical results obtained from simulations employing mesh models with and without symmetry boundary conditions at an airflow rate of approximately 450 slpm.

SETUP CONDITIONS AND RESULTS FOR SYMMETRY CHECK

REFERENCE VALUES

	Value:	
Air Density (kg/m^3)	0.98	
Air Viscosity (kg/s-m) $_{dynamic}$	1.85E-05	
Flow Area (m^2) $_{x\text{-}section}$	0.00968829	
Wetted Perimeter (m) $_{x\text{-}section}$	3.254596	
Hydraulic Diameter (m)	0.011907	
Turbulence Model	k-w standard $_{transitional}$	
Mass Flow Inlet (kg/s) $_{@\sim 450\ slpm}$	0.0098	(/ 4 if symmetry boundaries are used)
Pressure Outlet (Pa)	0	
Total Volume Cell Count $_{symmetry\ model}$	2773440	
Total Volume Cell Count $_{no\text{-}symmetry}$	8688816	

	Max Wall Y+ Value:	Avg Wall Y+ Value:	Max V_{outlet} (m/s):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:
Symmetry Model	2.7256086	0.9789276	2.2790601	2.506962	2.6569006
Full Section Model	4.9813242	1.8942908	2.2881179	2.468519	2.6524596
% Difference	NA	NA	0.395862	1.557343	0.167430

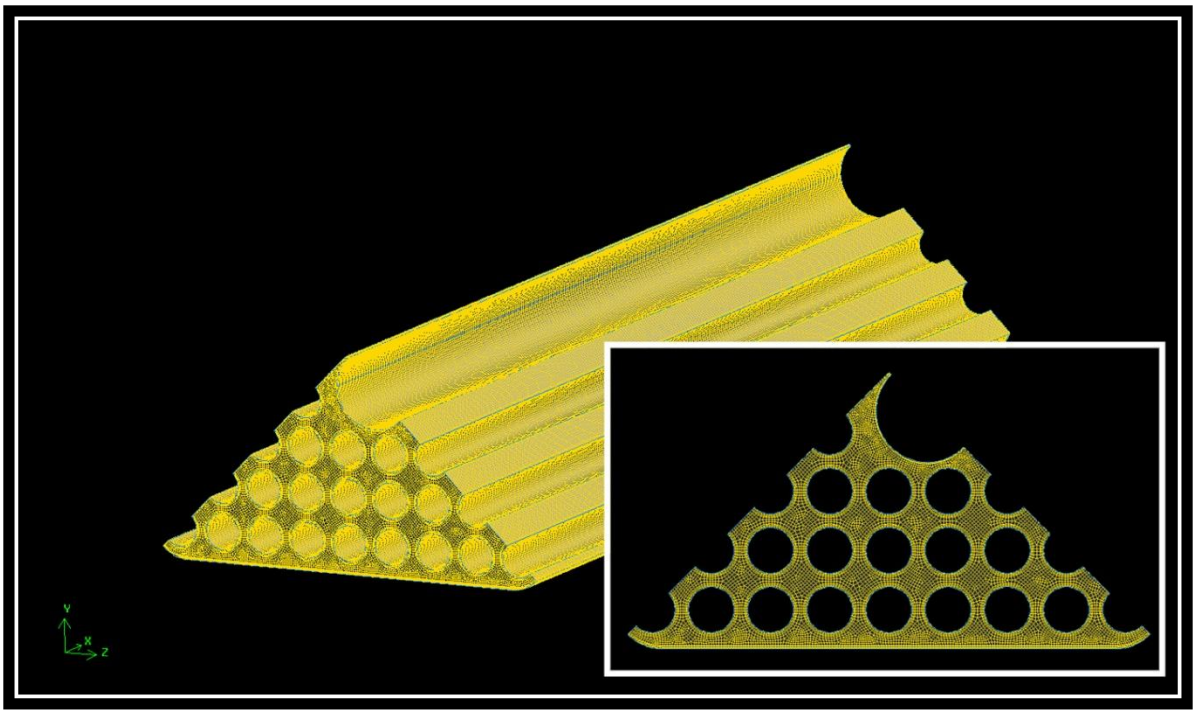


Figure 5.21: Isometric and cross-section views of mesh model with symmetry boundaries imposed.

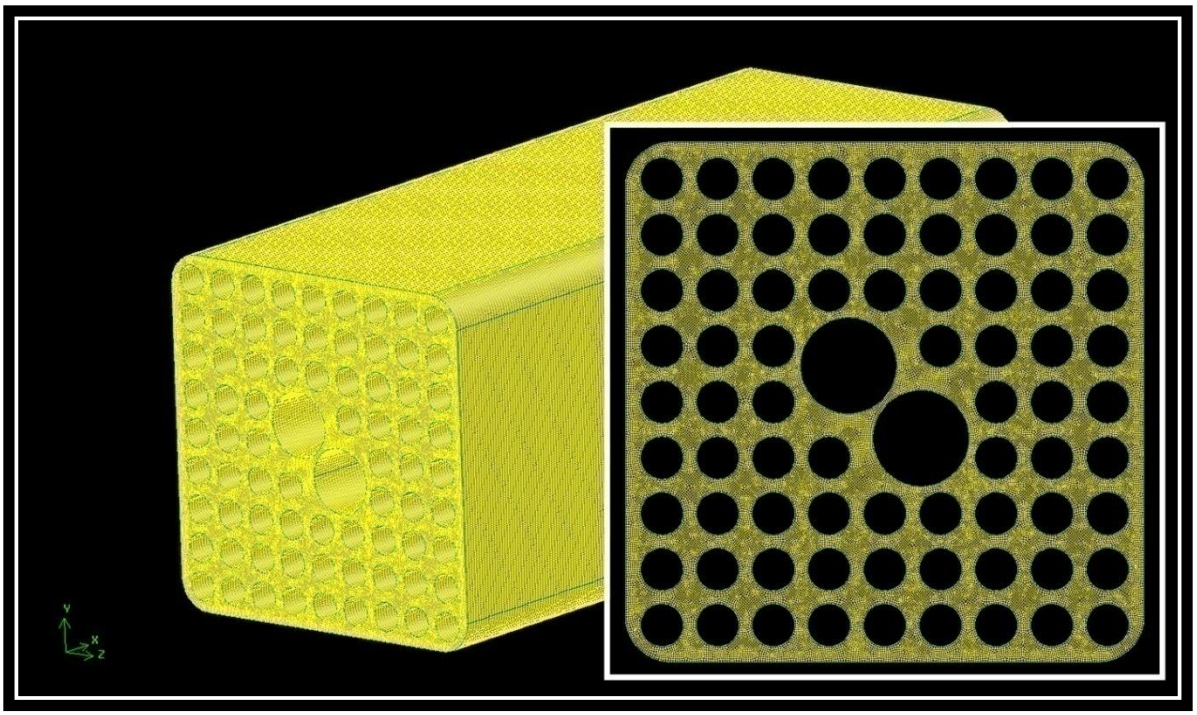


Figure 5.22: Isometric and cross-section views of mesh model without symmetry boundaries imposed.

The comparison of results in Table 5.3 provides a clear indication of the equivalence between both solution sets with parameter values differing at most by only a few percent, which consequently reflects the similarity in the resolved flow fields. This conclusion validates the proper implementation and location of symmetry boundaries within the computational domain for the type of flow phenomenon development modeled. In addition, the node clustering applied near symmetry boundaries and other areas of influence to the mirrored gradient patterns in the one-fourth size model reveals sufficient wall boundary values for the characteristic (y^+) turbulence parameter.

A second boundary condition explored follows the necessity for in-depth considerations akin to the symmetry boundary type in terms of adequate utilization and involves flow field solutions within a modeled domain that are anticipated to exhibit a

periodic, repeating nature alongside replicated cross-section geometry in the axial direction. This periodic boundary condition provides valuable insight into a number of domain flows in which minimal information is known with regard to the initial values and/or profile specifics of mass fluxes entering or exiting at system boundaries. Although an often iteratively stiff convergence problem to solve when utilizing this particular boundary type, the pressure losses for streamwise-periodic fluid flows and the reduction in grid sizes required to simulate fully-developed solutions (due to reduced physical domain lengths) with this boundary are extremely beneficial, especially for depicting potential airflow responses of the current assembly system analyzed in this research. Significance of the periodic flow assumption is illustrated in conjunction with solutions obtained from the physical experiments performed at SNL in the present and concluding discussions of Section 5.1, respectively.

Again, the fluid flow field is deemed periodic when the domain geometry and resolved pattern of thermal fluid flow behavior demonstrate a periodically repeating nature over a specified model length. The FLUENT CFD software is capable of modeling two different classifications of periodic flow that include cyclic (no pressure drop across periodic boundaries) and streamwise-periodic (constant pressure drop experienced across segment distinguished by periodic boundaries), in which the latter results in a fully-developed flow prediction that is most relevant to this research. Specifically, since the mass flow rate is known, the numerical solver computations are based under assumptions for the treatment of two particular parameters of fundamental importance: (1) periodic velocity and (2) local pressure gradient. The periodic velocity calculation is straightforward as periodicity implies that components of the velocity repeat in translational space equal to the distance between the periodic boundary planes.

On the other hand, the local pressure gradient is slightly more complicated since the value is comprised of both periodic and linearly-varying gradient components.

In short, the linearly-varying component of the pressure, which is responsible for the resultant fluid force acting in the momentum equations, is unknown and iteratively updated within this new subroutine calculation of the pressure correction step (FLUENT Inc., 2006). This is continued until the specified mass flow rate is acquired within sufficient limits imposed by the modeler. Obviously, as strictly dictated by the above definition of periodicity, there are no actual locations within the fuel bundle model that are appropriate for incorporating the periodic boundary conditions, which at minimum require a meshed flow domain that includes an assembly spacer component. However, the primary intent of this analysis is to compare and contrast flow-field developments across the fuel assembly bundle for both a uniform inlet flux profile and an assumed fully-developed state of flow. This serves as a future reference for the construction of subsequent larger-scale simulation models due to the lack of a specific attributable profile at the inlet without upfront consideration for the entire fuel assembly domain. Under such stated purpose and assumptions, the use of periodic boundary conditions across the fuel bundle model for simulating a fully-developed flow state is permissible.

Although a number of significant results and observations from these case study trials involving periodicity comparisons are withheld for concluding discussions at the end of Section 5.1, several key points of interest are illustrated in the periodic analysis of pressure losses presented below, which addresses airflows near both extremes of the flow rate range (*i.e.* approximately 450 and 150 slpm). Additional insight is visually provided by gauging differences in the velocity profile development from a uniform inlet flow, along a designated parallel-flow plane, with the periodic fully-developed profile at varying locations traversing the domain; velocity line profiles are plotted for comparison

using node values nearest the sectioning plane. These numerical and graphical results stemming from the preliminary periodic evaluations are displayed in Table 5.4 (with reference model values) as well as Figures 5.23 and 5.24 below.

Table 5.4: Numerical results from initial periodicity analysis at airflow rates of approximately 450 and 150 slpm.

SETUP CONDITIONS AND RESULTS FOR INITIAL PERIODICITY ANALYSIS

REFERENCE VALUES

	Value:
Air Density (kg/m^3)	0.98
Air Viscosity (kg/s-m) dynamic	1.85E-05
Flow Area (m^2) x-section	0.00968829
Wetted Perimeter (m) x-section	3.254596
Hydraulic Diameter (m)	0.011907
Viscous Model	k-w standard transitional
Mass Flow (kg/s) @ ~ 450 slpm	0.0098
Mass Flow (kg/s) @ ~ 150 slpm	0.00326667
Grid Study Domain Mesh	Trial 5
Boundary Types	Periodic, V_{inlet} - P_{outlet}

COMPARISON OF RESULTS

	Max Wall Y+ Value:	Avg Wall Y+ Value:	Max V_{inlet} (m/s):	Max V_{outlet} (m/s):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:
~ 450 (slpm)						
V_{inlet} - P_{outlet} Model	2.7277656	1.0032599	1.0739037	2.2814283	2.5109496	2.6612415
Periodic Model	1.4392555	0.9298193	3.026496	3.026496	0.0000683	2.0600924
~ 150 (slpm)						
V_{inlet} - P_{outlet} Model	1.4487410	0.5500959	0.3685142	0.9331216	0.7250044	0.7453485
Periodic Model	0.8097786	0.5225793	1.052777	1.052777	0.0000002	0.6424472

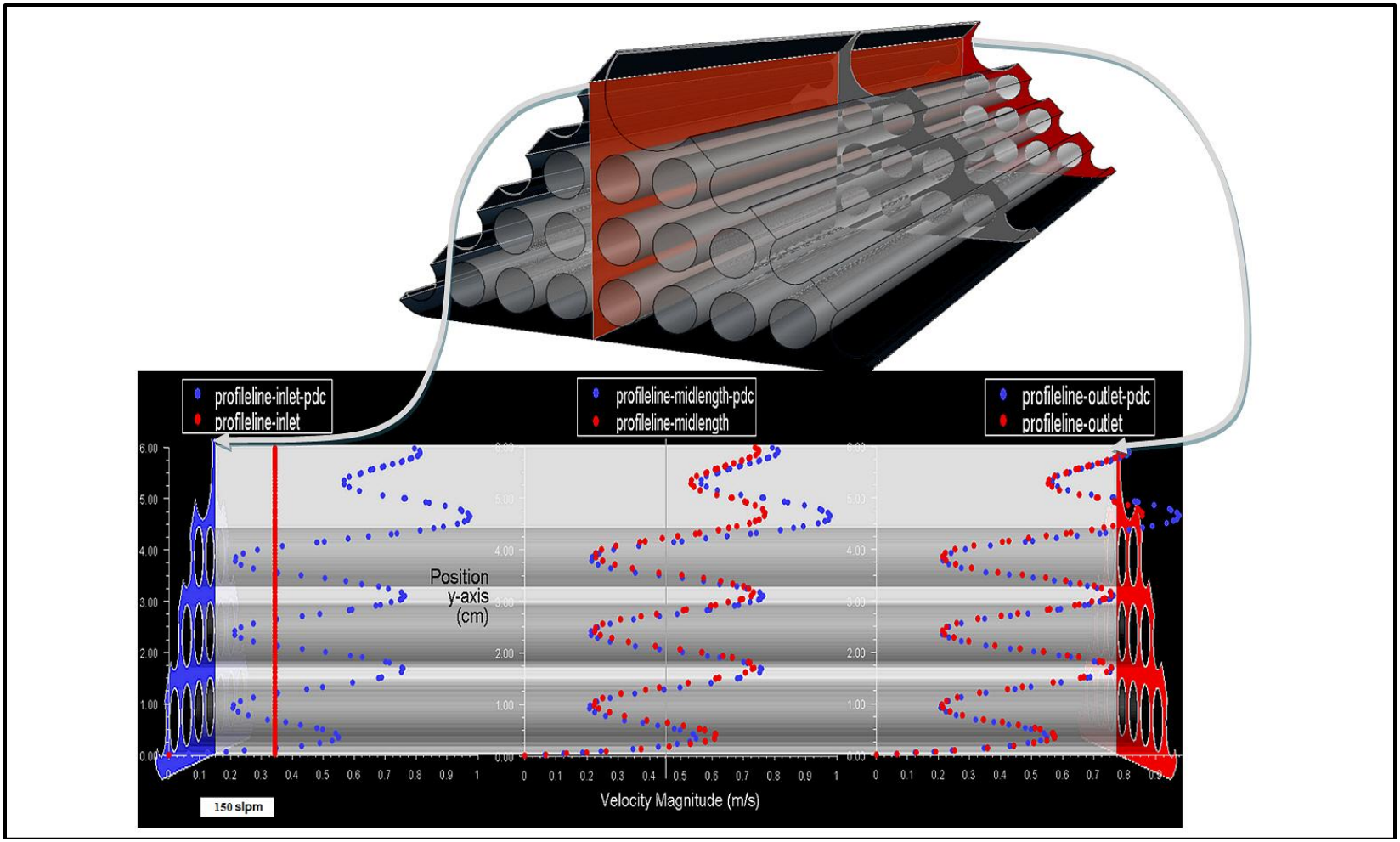


Figure 5.23: Line profile comparisons of velocity magnitude for periodic and uniform inlet trials at approximately 150 slpm.

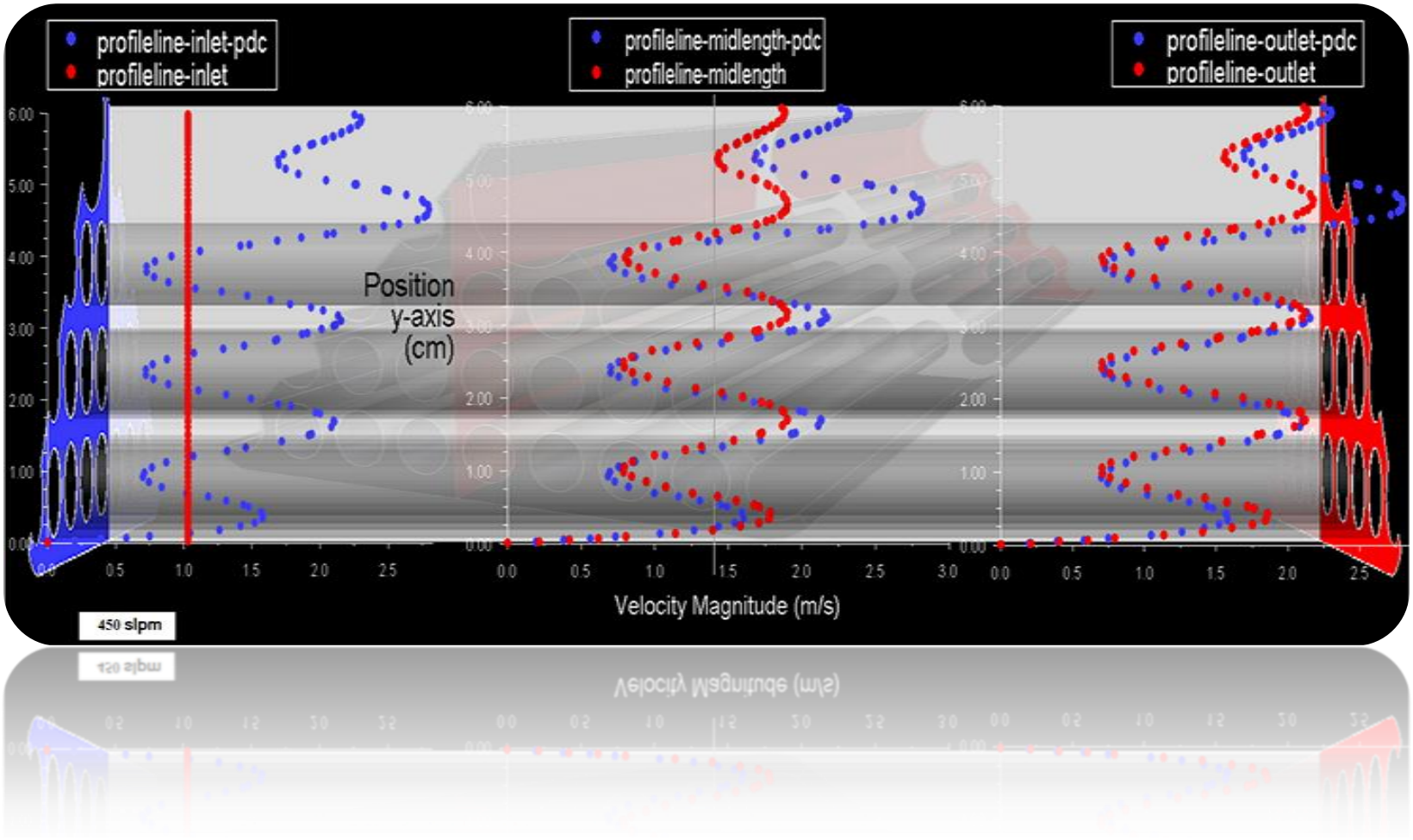


Figure 5.24: Line profile comparisons of velocity magnitude for periodic and uniform inlet trials at approximately 450 slpm.

The tabulated numerical solutions and line profile comparisons of velocity magnitude for periodic and uniform inlet conditions above are a substantially informative set of reference results with respect to the overall pressure losses anticipated as well as flow developments associated with each boundary type implementation. However, it is rather difficult to gain a detailed perspective of the developing velocity profile differences persistent throughout the entire cross-sectional area of flow across the domain by examining velocity line profiles in one plane at a single cross-section location. Such a thorough graphical representation requires the use of two- and/or three-dimensional contour plots for depicting these differences in airflow velocities based on solutions contained within two separate case/data files. Unfortunately, a built-in method that directly serves this function in the FLUENT CFD software does not exist. Although, with some basic knowledge of the C-programming language, it is possible to write and compile a user-defined function (UDF) into the FLUENT source code that affords this capability.

This added functionality is subsequently exploited to create detailed contour plots of cross-sectional velocity differences between periodic and uniform inlet simulations described in order to extend the comparison of resulting flow-field solutions. The UDF program code compiled into the FLUENT software is illustrated in Figure 5.25 followed by contour plots of resulting differences in the velocity field for both boundary condition cases in Figures 5.26 and 5.27 below.

```
#include "udf.h"

DEFINE_ON_DEMAND(u_to_udm0)
{

float u;

Thread *tc;
cell_t c;
extern Domain *domain;

thread_loop_c(tc,domain)
{
begin_c_loop(c,tc)
{
u=C_U(c,tc);
C_UDMI(c,tc,0)=u;
}
end_c_loop(c,tc)
}
}

DEFINE_ON_DEMAND(delta_u_to_udm1)
{

float delta;
float udmi0;
float u;

Thread *tc;
cell_t c;
extern Domain *domain;

thread_loop_c(tc,domain)
{
begin_c_loop(c,tc)
{
udmi0=C_UDMI(c,tc,0);
u=C_U(c,tc);
delta=udmi0-u;
C_UDMI(c,tc,1)=delta;
}
end_c_loop(c,tc)
}
}
```

Figure 5.25: UDF program code compiled into the FLUENT software for creating data file differences between velocities of two separate case/data solution files in which detailed contour plots of the variances are desired for analysis.

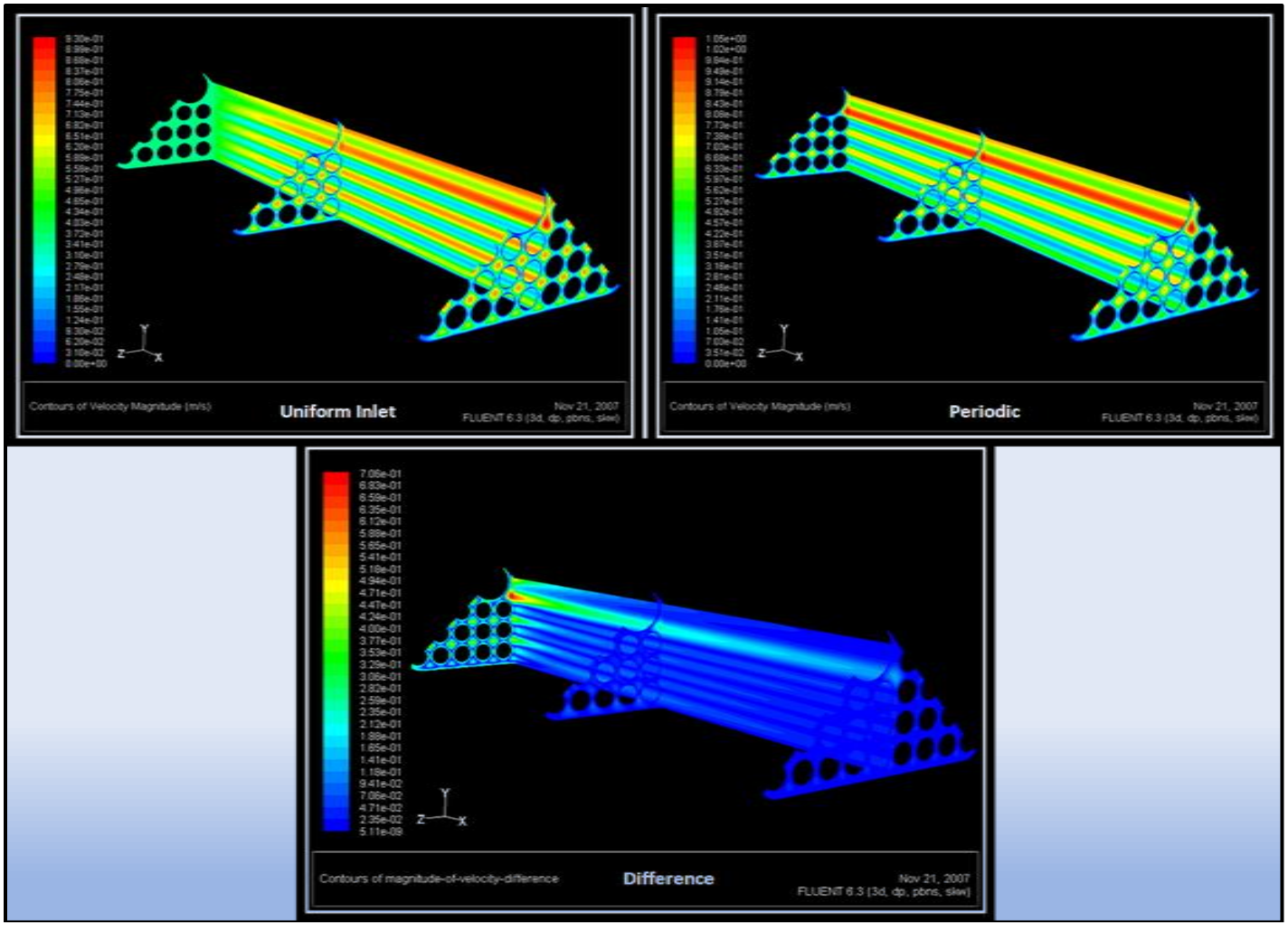


Figure 5.26: Uniform inlet and periodic results compared using interpreted UDF to create difference contours at ~ 150 slpm.

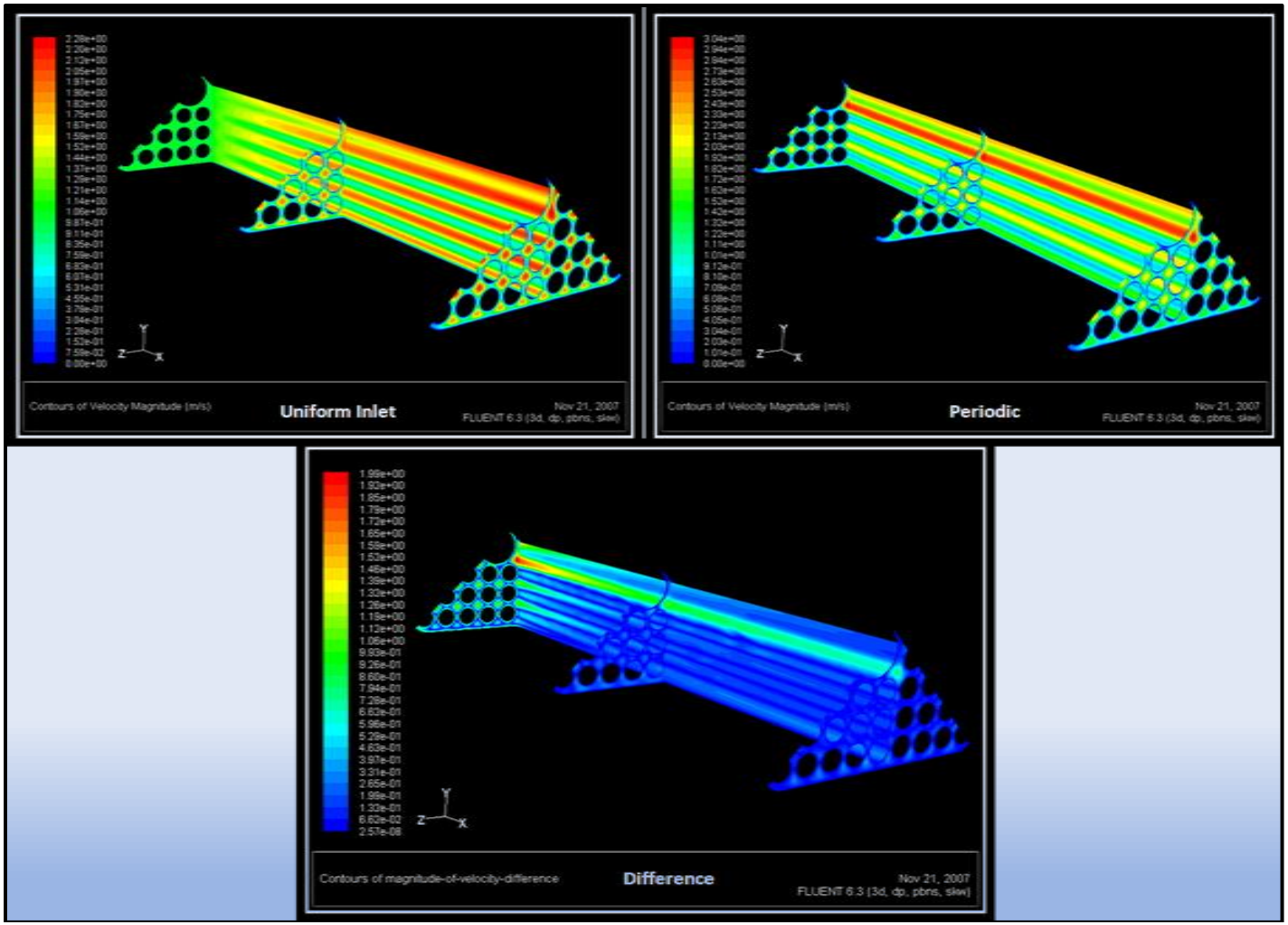


Figure 5.27: Uniform inlet and periodic results compared using interpreted UDF to create difference contours at ~ 450 slpm.

Application of the periodic boundary type to assess the range of plausible flow-field solution responses is a well-established practice under circumstances facilitated by inlet and/or outlet conditions that are otherwise unknown and too complex for estimation with widely attributed profiles arising from general wall-bounded flow analyses. Frequently, this knowledge of an assumed fully-developed flow state within the computational domain (regardless of the feasibility for such development) provides reference information essential in formulating a particular set of inlet and/or outlet flow conditions requisite for subsequent, larger-scale simulations of an extended model realm. Nonetheless, several key observations concluded from the preliminary comparative analysis above for flow-field developments involving both uniform velocity inlet and periodic boundary conditions at airflow rates of approximately 150 and 450 slpm are briefly summarized in the bulleted list that follows.

- An extensive range of potential pressure losses corresponding to the varying degrees of practical flow development for airflows traversing the fuel bundle domain is encapsulated by the resultant flow fields from uniform velocity and periodic boundary conditions imposed at the inlet. The breadth of these pressure loss values is limited at the higher extent by specification of a uniform inlet velocity and the lower extent by conditions of periodicity (for a designated assembly airflow rate).
- Gauging the magnitude of importance associated with the accurate allocation of a correct inlet boundary profile is realized through direct observation of substantial differences in the predicted pressure losses at both flow rates for each boundary type employed.

- Without modeling the fuel assembly domain from the inlet onwards (in the downstream direction), identification of a sufficiently accurate inlet boundary profile is viable only through direct comparison with, and in consideration of, the experimental pressure loss measurements from the fuel assembly hydraulic analysis conducted by SNL. This subject is addressed at greater length alongside the concluding discussion of results in Section 5.1.7 below.
- Another noteworthy deduction afforded by this analysis is derived from the contour plots of velocity field differences illustrated in Figures 5.26 and 5.27 for solutions of both inlet boundary-type specifications. These contour plots clearly indicate an appreciable similarity between cross-sectional velocity profiles of the fully-developed and progressively developing flow fields for periodic and uniform inlet boundary conditions near the model outlet; slight variances are limited to a minimal interstitial flow area adjacent to the central water rod surfaces. This implies that an established state of fully-developed flow is highly probable at some point across the fuel bundle length regardless of most upstream and downstream effects anticipated from assembly spacer components or the airflow rate specified.
- In addition, the existence of a periodic, fully-developed flow field is further exemplified by including the following two considerations: (1) the fuel bundle domain represented in this model only constitutes a centered, two-thirds length segment of the actual fuel bundle length spanned by two successive spacers and (2) velocity profiles immediately downstream of the spacer components possess a near certain level of increased flow development in comparison to the uniform inlet boundary condition utilized above. As a result, employing the assumption of periodicity across a modeled domain encompassing an entire fuel bundle and

spacer segment of single-span length is heavily expected to carry meaningful significance in the hydraulic analysis of fully-populated sections of the assembly flow volume.

5.1.4 Analysis of Viscous Models

The ensuing analysis briefly focuses on the general implications associated with several predominant viscous model implementations and available treatment options as well as the verification of appropriate application conditions and capability of providing accurate solutions to resolved flow fields of direct relevance to this research. In particular, the motivation for conducting this viscous model assessment is threefold including affirmation of the following: (1) adequate capture of important flow-field characteristics with proper viscous model selection; (2) correct setup as well as utilization of corresponding model options; and (3) sufficient boundary-layer cell distribution and resolution of mesh for the viscous model employed (*i.e.* y^+ cell values). Operation of the fuel assembly apparatus within the laminar flow regime with Reynolds numbers spanning 70 to 900 (Durbin and Lindgren, 2005) provides the primary criterion for initial selection of the viscous models used in this study.

A second consideration is based on the desire to maintain congruent mesh requirements among the models chosen for reasons of a practical nature in performing a meaningful comparative analysis. Fortunately, situations involving low-Re bounded flows where accurate measurements of differential pressure (resulting from small-scale viscous losses near walls) are necessary across domain extents also dictate grid resolutions of similar constraint for the suitable viscous model implements available in the FLUENT CFD software. As a result, the viscous models deemed most appropriate

for this research endeavor include the Spalart-Allmaras (SA), standard k-ω transitional, k-kl-ω (beta), and general laminar viscous models. The fundamental theory, basic equations, and relevant applications associated with the turbulence model options are briefly introduced in Section 2.1.3 above; however, further modeling details and evidence supporting the adequacy of these selected viscous models is available in the literature elsewhere (Wilcox, 1998; Walters and Leylek, 2004; FLUENT Inc., 2006).

Aside from the laminar viscous model, a suggested guideline for the wall unit value of near-wall cells common to each of the aforementioned turbulence models is ($y^+ \approx 1$) with an emphasis on avoiding ($y^+ > 5$) as stated in the FLUENT user's guide (FLUENT Inc., 2006). For computations employing the laminar model option, certain circumstances permit the use of a less-stringent grid resolution (as compared to the turbulence models above) depending upon the outlined intent of the simulation(s) performed. In order to assess the influence of grid resolution on imparted flow-field solutions stemming from case trial setups coupled with the laminar viscous model selection, results are obtained utilizing both the default high-resolution mesh specified for the turbulence models (*i.e.* trial 5 grid) and a coarser mesh of moderately reduced cell count. An evaluation of these additional results is incorporated into the analysis below.

Assuming the bulk flow-field response is characterized by behavior attributable to the laminar flow regime, an equivalent set of solutions is anticipated for each of the viscous modeling options examined (within acceptable limits); the expectation is further predicated on retaining the correct submodel implementation and proper mesh resolution in all respective simulations conducted. The airflow through the fuel bundle domain is designated at a higher-end value of the flow rate range (approximately 450 slpm) for this analysis since the greatest discrepancies among solutions for each viscous model option involved in the study are projected to coincide with the higher airflow rate values.

Results for several principal flow-field parameters including y^+ (if applicable), outlet velocity, and pressure loss are accessed for general comparison of the viscous model solutions as well as the capability of individual model implements to accurately resolve flow domains of relevant significance to the current research. These numerical results and associated reference model values for the simulation trials performed under two different boundary condition combinations are provided in Table 5.5 below.

Table 5.5: Reference values and numerical results from the analysis of viscous models at approximately 450 slpm.

SETUP CONDITIONS AND RESULTS FOR VISCOUS MODEL ANALYSIS

REFERENCE VALUES

	Value:
Air Density (kg/m^3)	0.98
Air Viscosity (kg/s-m) $_{dynamic}$	1.85E-05
Flow Area (m^2) $_{x\text{-}section}$	0.00968829
Wetted Perimeter (m) $_{x\text{-}section}$	3.254596
Hydraulic Diameter (m)	0.011907
Viscous Models	k-w standard $_{transitional}$, k-kl-w (beta) , Spalart-Allmaras $_{vorticity}$, laminar
Mass Flow (kg/s) $_{@\sim 450\ slpm}$	0.0098
Uniform Inlet Velocity (m/s) $_{@\sim 450\ slpm}$	1.032174
Grid Study/Other Meshes Used	Trial 5 and Coarse Variant
Inlet/Outlet Boundary Types	M_{inlet}, V_{inlet} - P_{outlet}

COMPARISON OF RESULTS

Viscous Model	Inlet BC Used:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Max V_{outlet} (m/s):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:
k-kl-w (beta)	Mass Flow	2.6730769	0.9971889	2.2956419	2.4694587	2.6264604
SA $_{vorticity}$	Mass Flow	2.7281132	1.0096437	2.2266252	2.5339194	2.6684585
k-w standard $_{transitional}$	Mass Flow	2.7256086	1.0023551	2.2790601	2.5030020	2.6569006
k-w standard $_{transitional}$	Uniform Velocity	2.7277656	1.0032599	2.2842830	2.5109496	2.6612415
Laminar $_{course\ grid}$	Uniform Velocity	NA	NA	2.1529982	2.1500700	2.2531245
Laminar $_{fine\ grid}$	Uniform Velocity	NA	NA	2.3013608	2.5102782	2.6529684

A quick review of the numerical results compared in Table 5.5 above leads to the deduction of three points of explicit interest with regard to the implementation of viscous modeling options. Foremost, a clear consistency across the resolved flow fields is observed between all turbulence models with near equivalent solution sets acquired for the parameter values compared. Combined with maximum and average cell y^+ values for each viscous model that sufficiently fall within the suggested guideline limits, strong support for the correct implementation of these turbulence model options and the capability of all to resolve related flow fields accurately is established.

Shifting attention toward considerations for the laminar model implement and the influence of grid resolution on flow-field solutions obtained, reveals a second point of viscous modeling interest. As indicated by the large discrepancies between the parameter solutions obtained on the same high-resolution mesh used for the turbulence models and a coarser mesh of moderately reduced cell count, the level of grid refinement in near-wall boundary areas substantially affect the flow-field solution imparted by the laminar viscous model under the stated airflow conditions. The inappropriate notion that grid resolution requirements are inherently less stringent for simulations utilizing the laminar viscous model to resolve flows in the laminar regime, as compared to low-Re turbulence model counterparts, is commonly assumed without adequate deliberation. This assumption is especially detrimental in cases involving low-Re, wall-bounded flows in which accurate measurements of differential pressure due to small-scale viscous losses at boundary walls are critical, as similarly pertains to the current research.

In short, the accuracy of flow losses predicted by the laminar viscous model relies exclusively on the resolution of near-wall velocity gradients since the wall shear stress is defined by the normal velocity gradient at the wall. Therefore, a near-wall mesh resolution similar to that imposed for the above turbulence models is likely necessary in

order to obtain accurate flow-field solutions for any comparable airflow conditions using the laminar model selection. Exploiting coarser mesh resolutions with reduced grid file sizes in conjunction with the laminar viscous model is certainly unsubstantiated for the domain flows assayed in this research.

In conclusion, the remarkable similarity between parameter solution values resolved on the high-resolution mesh coupled with the laminar model option and those attained from all evaluated turbulence models affords reasonable justification of the preceding point. Although each of the viscous models examined exhibit a legitimate capability to provide accurate flow-field solutions under domain conditions of particular relevance to this research as demonstrated by the comparison of results in Table 5.5, only two of these model options are elected for a more in-depth assessment and final comparison of results discussed in Section 5.1.7 below. Obviously, one of these selections includes the laminar viscous option due to the potential benefit of significantly reduced resource requirements associated with computational experiments employing this viscous model.

The turbulence viscous model chosen for comparison that is deemed most appropriate among the three available options in the current research is the standard k-ω transitional model. This choice is based on a number of different considerations, but three chief aspects supporting the decision include the following (FLUENT Inc., 2006): (1) the minute body of work validating the recently introduced k-kl-ω (beta) turbulence model under a range of varying flow conditions as well as the precise beta adaptation incorporated into the FLUENT software is a significant drawback; (2) like other one-equation models, concerns arise with regard to the Spalart-Allmaras model and its suitability for domain flows in which notable changes in length scale are present as potentially relevant across multiple components within the fuel assembly; and (3) the

validity of the standard k-ω model for accurately resolving low-Re, wall-bounded flows is supported by a sizeable extent of benchmark assessments as opposed to the relatively new k-kl-ω (beta) turbulence model.

5.1.5 Boundary Conditions and Turbulence Parameters

The primary intent for this brief section insert is simply to serve as a placeholder for acknowledgement of concurrent considerations concerning various boundary condition types and values addressed throughout aforementioned research assessments in this chapter. A thorough analysis of relevant boundary specifications is a critical component for demonstrating a complete understanding of the modeled flow domain including the potential range of system responses, modeling implications associated with imposed boundary types, and justification for the conditions implemented in obtaining the final flow-field solutions. The conglomeration of boundary conditions utilized in the simulation case trials above span multiple combinations of both velocity and mass-flow inlet with pressure and mass-flow outlet boundary types as well as uniform and fully-developed inlet profile specifications. Moreover, the imparted knowledge and benefits afforded by both periodic and symmetry boundary treatments are initially gauged and fully realized with the final comparison of results in Section 5.1.7 below.

On a final note, the computational trials incorporating any turbulence viscous model further necessitate initial guesstimates for one or more corresponding turbulence parameters at inlets and outlets of the flow domain. Although final calculated solutions are generally insensitive to these transported turbulence quantities across model boundaries, it is important to ensure that specified values are not contaminating the solution or impeding convergence. Therefore, since the intensity and hydraulic diameter

specification method is universally implemented in all relevant simulations with a turbulence viscous model (the hydraulic diameter value always known), the turbulence intensity at both the inlet and outlet are varied within a physically realistic range of values in order to assess any effects on the predicted flow-field solutions. As expected, the variances in turbulence intensity resulted in solution differences of negligible significance.

5.1.6 Residual Criteria and Convergence Monitors

The last assessment conducted in an effort to establish the legitimacy of both computational model and corresponding solutions addresses the issue of incomplete iterative convergence error through validation of the residual convergence criteria and convergence monitors employed in determining solution convergence. A general guideline followed for judging the convergence of simulations performed throughout this research requires that scaled residuals for each flow equation reach a specified decrease in order of magnitude prior to the assertion of solution convergence. Setting sufficient, but not overly tight residual criteria for the flow equations affords a proper balance between computational cost and minimization of error due to incomplete iterative convergence.

In order to gauge adequate limits for the residual convergence criteria over a reasonable range of airflows expected to develop across the fuel bundle domain, two simulation cases are setup with flow rates stemming from the higher- and lower-end range of anticipated values (approximately 450 and 150 slpm, respectively). These trials implement the uniform velocity inlet and pressure outlet boundary conditions as well as the standard k-ω transitional viscous model on a high-resolution mesh. The simulation at

each flow rate is calculated until various residual criterion values for the flow equations are achieved at which point the pressure loss is evaluated along with plots of residual behavior, mass imbalance, and AWA static pressure difference across the fuel bundle model for evidence of solution convergence. Additional convergence monitors such as the mass imbalance and AWA static pressure difference used in these computational experiments are critical for any judgment of convergence since a number of situations arise in which the normalized or scaled residuals include unknown initial influences from the solution initialization process that misrepresent or prevent achievement of certain reductions in the residual values.

The residual criterion levels for each equation are varied between $1x10^{-3}$ and $1x10^{-5}$ in this analysis, which span an acceptable range of potential convergence levels necessary for accurate resolution of the flow-field solutions. Ultimately, the results of this assessment reveal that a minimum residual criterion of $1x10^{-4}$ or tighter (for all equations) is sufficient for obtaining accurate solutions across the range of airflow rates studied. The plots of residual behavior, mass imbalance, and AWA static pressure difference across the domain with respect to iteration number are graphically illustrated at each flow rate in Figures 5.28 and 5.29 below for the entire set of criterion values reviewed.

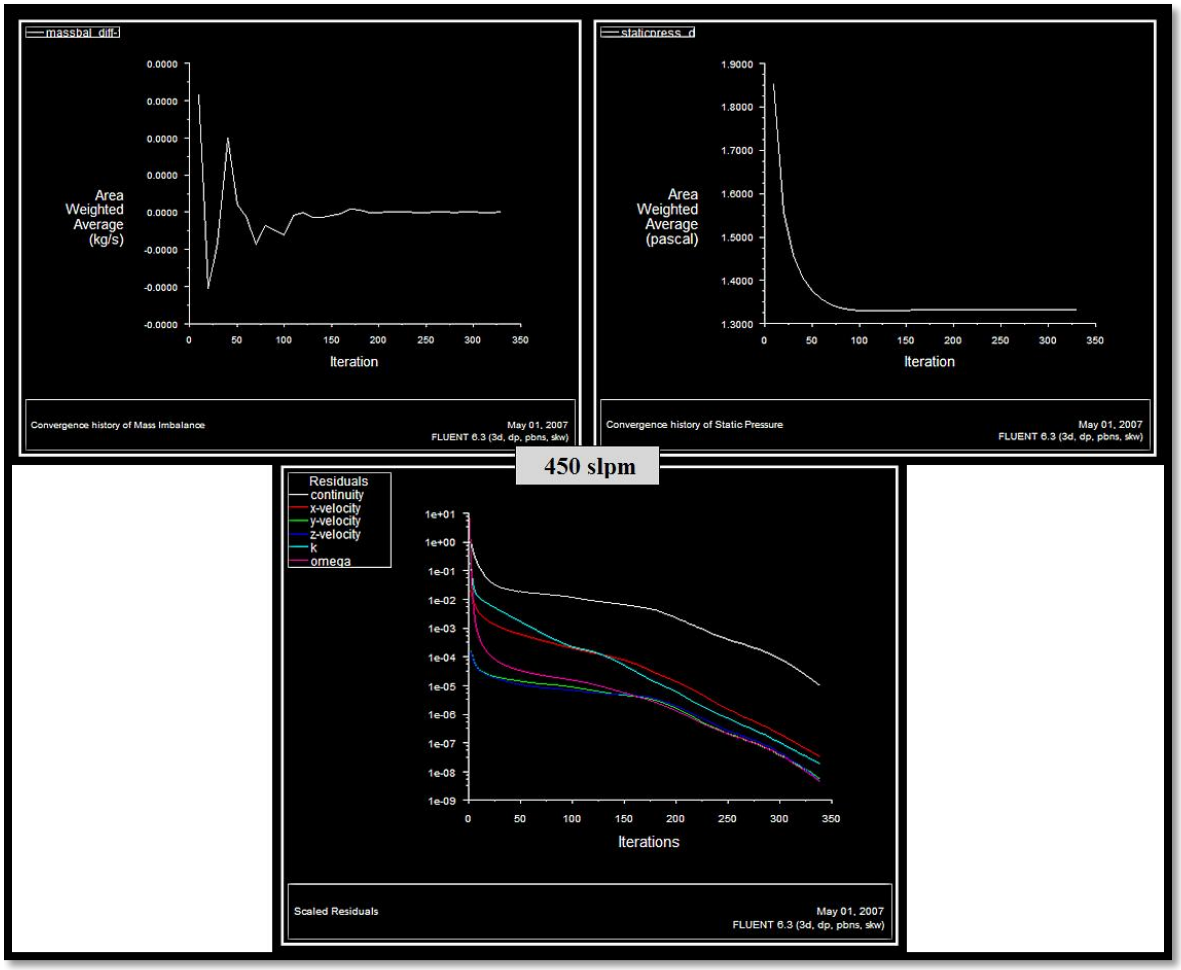


Figure 5.28: Plots of scaled residuals, mass imbalance, and AWA static pressure difference across model domain for convergence study at an airflow rate of approximately 450 slpm.

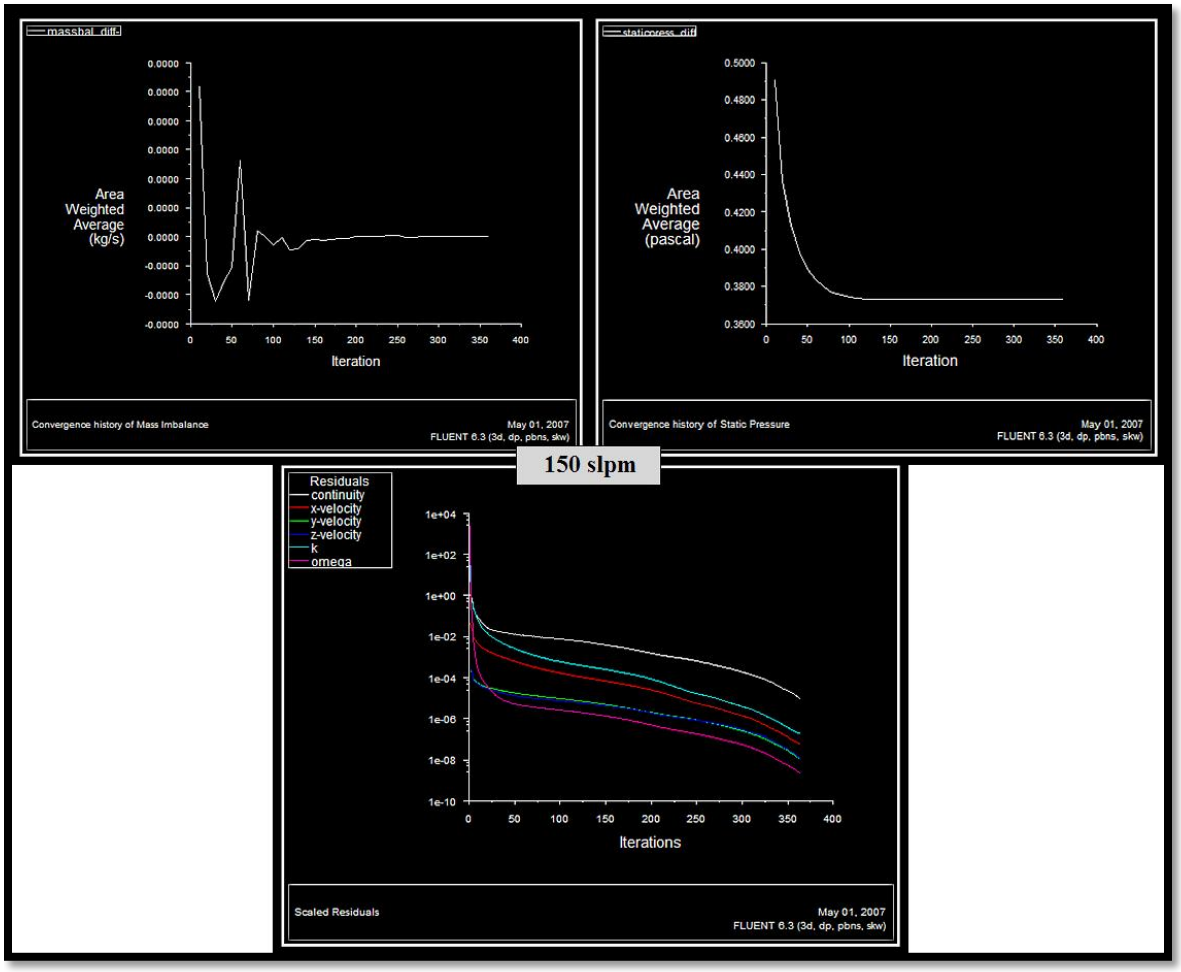


Figure 5.29: Plots of scaled residuals, mass imbalance, and AWA static pressure difference across model domain for convergence study at an airflow rate of approximately 150 slpm.

At minimum, two additional convergence monitors are selected in the form of integral values that are tracked throughout the entirety of a simulation. The first monitor, which is utilized in all case trials of known mass flow rate, requires the net mass imbalance be less than 0.1% of the net flux through the domain. As for the second monitor tracked, if a uniform parameter value is specified at the inlet boundary then an AWA pressure measurement of some designation across the modeled domain is incorporated. However, if periodic flow conditions are enforced, the second monitor

implemented for tracking is the periodic pressure gradient statistic. Convergence is suggested in both of these secondary monitors by observing negligible changes in the target value over a reasonable number of solver iterations.

5.1.7 Final Comparison of Results

The results presented and discussed in this section are selected based upon a thorough understanding of the knowledge afforded by completing the numerous case studies outlined in Figure 5.2 above and after detailed consideration for the corresponding solutions obtained. As previously emphasized, the primary motivation for conducting these assessments is threefold: (1) validate accuracy of computational results by sufficiently addressing numerical uncertainty according to guideline criteria summarized in Section 2.4 above; (2) identify general resource requirements and obstacles that exist early in the model development process, which pose future potential problems for progressively larger-scale domain models; and (3) establish some level of support for the direction and implementation of subsequent large-scale simulations undertaken without current, adequate justification of employed assumptions due to constraints associated with an original CFD modeling effort. These initial research trials involving a smaller segment of the actual flow domain within the experimental fuel assembly are also intended to contribute a modest set of supplementary hydraulic analysis solutions for corroboration of MELCOR SFP model calibrations at SNL.

The final comparison of results embraces a collection of numerical flow-field data that effectively reflects a span of probable system flow responses for the fuel bundle domain, which impart meaningful revelations concerning the development of succeeding larger-scale models using a combination of simulation setups, solver options, and

convergence criteria as appropriately allocated based on the above assessments. In particular, specific pertinent features common to all final computational trials include a mesh characterized by the trial 5 grid refinement, insertion of symmetry boundary planes, and a convergence criterion of at least $1x10^{-4}$ for the scaled residuals of all equations accompanied by additional convergence monitors of mass imbalance and a second boundary-condition dependent parameter (as noted in Section 5.1.6). These aspects are derived from conclusions drawn in the corresponding grid independence, symmetry, and convergence criteria studies, respectively.

The analysis of viscous model options reveals preliminary evidence supporting the capability of both laminar and standard k-ω transitional viscous models to resolve the relevant flow fields associated with traversing fuel bundle airflows. Thus, the capability of both viscous treatment options is further elucidated by examining the accuracy of solutions obtained from each model implement over an extended array of flow rates in the final comparison of results below. Applicable boundary type specifications most appropriate for capturing the desired range of potential flow-field responses include the periodic and combination velocity inlet with pressure outlet set of boundary types; noting the velocity inlet boundary entails joint options for stipulating either a uniform (location-constant) or profile arrangement of velocity values across the inlet boundary.

Considerations for exploiting the profile boundary condition at a velocity inlet is realized in the following investigations through attainment of velocity values at a cross-sectional plane (normal to the flow) from resolved flow-field solutions with periodic boundary conditions imposed. Each fully-developed velocity profile is attached to a velocity inlet boundary in conjunction with a pressure outlet condition and subsequently solved for a converged solution. These velocity profile assessments clearly serve as an additional verification of solution convergence in the periodic flow simulations; however,

the full significance of this boundary condition specification at velocity inlets is not apparent until extended model developments of Section 5.2 are discussed.

Finally, pressure loss results from the computational trials are compared with experimental measurement values obtained indirectly through specified pressure loss calculations and findings presented in the final hydraulic analysis report by SNL (Durbin and Lindgren, 2005). The underlying methodology associated with this hydraulic analysis is sufficiently introduced in Chapter 3 including complete designations for ambient/air properties, viscous and inertial loss coefficients, assembly/component dimensions, and pressure loss formulas as implemented in the SNL hydraulic analysis of the SFP pressure drop experiments, which are required for calculation of pressure loss values at compared flow rates below. Due to modest differences in the manufacturer dimensioned drawings used to construct the computational assembly model and those reported by SNL as representing the actual experimental assembly, slight variances exist between the hydraulic diameter and flow area values characterizing the fuel bundle domain employed in each analysis. Although minor variances in velocity for an equivalent flow rate combined with marginal difference in hydraulic diameter impose a degree of pressure loss divergence as an end consequence of these discrepancies, the airflow velocities are not equated in concern for a more-direct comparison of these assessments owing to the importance of maintaining consistent mass flow rate comparisons across all domains.

However, equating the fuel bundle lengths gauged under both computational and experimental model constructions is paramount to any practical solution comparison of meaningful consequence. As repeatedly emphasized, the fuel bundle model applied in all aforementioned simulations of this chapter constitutes only a section of the actual bundle run length between two full-populated spacer components. The centered, two-thirds

equivalent length model of a fully-populated bundle run is formulated under a purely hypothetical set of conditions premised on inspection of bundle flow developments with a majority removal of upstream and downstream spacer effects by negating a total length of approximately fourteen hydraulic diameters from the bundle run domain at adjacent spacer boundaries.

Therefore, in order to compare pressure losses for an equivalent length domain based on the experimental hydraulic analysis conducted by SNL, the bundle segment length (L) in Equation (3.9) is set equal to the reduced computational model length. Obviously, this compels an assumption that viscous pressure losses are somewhat proportional across the domain length and reasonably reflect actual viscous losses over a two-thirds length segment of the fuel bundle. Furthermore, since inertial pressure losses are quantified by a single incurred loss value assigned to a component entirety and not length, the quantity is not susceptible to appropriation of a modified domain length and thus assumed to remain a reasonable estimate of inertial losses even with the reduced fuel bundle length.

In summary, despite a number of model discrepancies and a few modest concerns with regard to appropriate application of SNL hydraulic analysis results outside original intent, the influences on solutions obtained for comparison are assumed tolerable and the pressure loss values are anticipated to reflect actual measurements within a 10% margin of error. For the final judgment of results, five different airflow rates are impartially selected at near uniform intervals and roughly confined between 100 and 500 slpm. The important references, initial values, and setup conditions for both computational and experimental trials as well as the SNL hydraulic analysis calculations and comparison of final numerical results for the fully-populated bundle run model are all included in Tables 5.6 and 5.7 below. In addition, considerable postprocessing of graphical results for

parameter values of particular interest from selected computational trials and airflow rates are illustrated in Figures 5.30 through 5.42, which follow. Although extensive effort is expended in an attempt to provide high-end graphical postprocessing for all simulations, the immense number of parameter solutions and model setups analyzed, compounded by substantial computational resource and research scheduling costs, preclude this as a truly viable consideration. Therefore, focus is directed more-heavily upon flow rates and trial setups characterized by values or options at either end of the evaluated range of concern, which impart the most meaningful results to begin.

Table 5.6: Final comparison of results for fuel bundle model with SNL hydraulic analysis data and setup conditions (1 of 2).

[FINAL RESULTS AND SETUP CONDITIONS FOR FUEL BUNDLE RUN (FULLY-POPULATED) MODEL WITH SNL EXPERIMENTALLY-BASED HYDRAULIC ANALYSIS DATA]

REFERENCE VALUES (COMPUTATIONAL)

	Value:
Air Density (kg/m^3)	0.98
Air Viscosity (kg/s-m) $_{dynamic}$	1.85E-05
Flow Area (m^2) $_{x\text{-}section}$	0.00968829
Wetted Perimeter (m) $_{x\text{-}section}$	3.254596
Hydraulic Diameter (m)	0.011907
Viscous Model 1	k-w standard $_{transitional}$
Viscous Model 2	Laminar
Grid Study Domain Mesh Used	Trial 5
Face Cell Count $_{x\text{-}section}$	8667
Total Volume Cell Count $_{domain}$	2773440
Total Node Count $_{domain}$	3030882
Inlet/Outlet Boundary Types	V_{inlet} - P_{outlet}, Periodic
Pressure Outlet Gauge Pressure (Pa) $_{outlet}$	0

REFERENCE VALUES (EXPERIMENTAL - SNL HYDRAULIC ANALYSIS)

	Variable:	Value:	Notes:
Hydraulic Diameter (m)	D_H	0.0119	
Domain Length (m)	L	0.320886667	Set To FLUENT Model !!!
Laminar Coefficient	S_{LAM}	77	Hydraulic Coefficient 1
Form Coefficient	$\sum k$	0.84	Hydraulic Coefficient 2
Viscosity (kg/s-m) $_{dynamic}$	μ	1.85E-05	
Air Density (kg/m^3)	ρ	0.98	
Flow Area (m^2) $_{x\text{-}section}$	Ac	0.00979	

Flow Rate (slpm)	Bundle Velocity (m/s)
~ 450	1.021450
~ 380	0.851209
~ 300	0.680967
~ 230	0.510725
~ 150	0.340483

Bundle velocities are slightly different due to a dissimilar flow area used by SNL as compared with numerical experiments.

SNL HYDRAULIC ANALYSIS FORMULAS AND CALCULATIONS

$$\Delta P_{Total} = a_2 V^2 + a_1 V$$

Where: $a_1 = \frac{S_{LAM} \cdot L \cdot \mu}{2 \cdot D_H^{\,2}}$ $\quad a_2 = \frac{\rho \cdot \sum k}{2}$

Using these formulas: $a_1 =$ 1.613950488 $\quad a_2 =$ 0.4116

Accordingly: $\Delta P_{Total} = (0.4116) V^2 + (1.614) V$

COLOR KEY - BOXED SOLUTIONS

	Solution Value / Input Data
	Not Available In FLUENT For Model / Data Type
	Simulation Not Performed / Deemed Unnecessary

Note: Total subscript label is specific to minor plus major losses, not total pressure as given in FLUENT.

Final SNL Results: (Based on hydraulic analysis data from experimental results)	Flow Rate (slpm):	Pressure Loss $(Pa)_{static}$:
	~ 450	2.078018
	~ 380	1.672036
	~ 300	1.289912
	~ 230	0.931647
	~ 150	0.597240

NUMERICAL PARAMETER SOLUTIONS FOR MODELED CONDITIONS OF SIGNIFICANT INTEREST AND FINAL COMPARISON OF RESULTS

Flow Rate: ~ 450 slpm

								Data Comparison		
								(Computational)	(Experimental)	(Gauge)
Viscous Model	Inlet BC:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Avg Wall τ_w (Pa):	Max V_{outlet} (m/s):	Avg $P_{dynamic@outlet}$ (Pa):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:	Pressure Loss $(Pa)_{static}$:	% Difference:
k-w standard $_{transitional}$	Uniform Velocity	2.727766	1.109578	0.022271	2.281428	0.665038	2.510950	2.661242	2.078018	28.066343
k-w standard $_{transitional}$	Periodic	1.439256	1.027886	0.018763				2.060092	2.078018	0.862624
k-w standard $_{transitional}$	Periodic Velocity Profile	1.699554	1.048058	0.019594	2.264857	0.647242	2.102478	2.065809	2.078018	0.587515
Laminar $_{fine\text{-}mesh}$	Uniform Velocity			0.022259	2.301361	0.667995	2.510278	2.652968	2.078018	27.668219
Laminar $_{fine\text{-}mesh}$	Periodic			0.018458				2.058620	2.078018	0.933480
Laminar $_{fine\text{-}mesh}$	Periodic Velocity Profile								2.078018	

Table 5.7: Final comparison of results for fuel bundle model with SNL hydraulic analysis data and setup conditions (2 of 2).

Flow Rate: ~ 380 slpm								Data Comparison		
								(Computational)	(Experimental)	(Gauge)
Viscous Model	Inlet BC:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Avg Wall τ_w (Pa):	Max V_{outlet} (m/s):	Avg $P_{dynamic@outlet}$ (Pa):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:	Pressure Loss $(Pa)_{static}$:	% Difference:
k-w standard$_{transitional}$	Uniform Velocity	2.454968	1.001972	0.018139	1.984508	0.466542	2.031141	2.140496	1.672036	28.017321
k-w standard$_{transitional}$	Periodic	1.308446	0.922039	0.015132				1.637867	1.672036	2.043550
k-w standard$_{transitional}$	Periodic Velocity Profile	1.371851	0.930348	0.015426	1.948653	0.464062	1.658927	1.640665	1.672036	1.876251
Laminar$_{fine\text{-}mesh}$	Uniform Velocity			0.018148	1.987946	0.468270	2.031504	2.134852	1.672036	27.679804
Laminar$_{fine\text{-}mesh}$	Periodic			0.015055				1.636305	1.672036	2.136975
Laminar$_{fine\text{-}mesh}$	Periodic Velocity Profile			0.014958	2.537136	0.486461	1.624643	1.625934	1.672036	2.757236

Flow Rate: ~ 300 slpm								Data Comparison		
								(Computational)	(Experimental)	(Gauge)
Viscous Model	Inlet BC:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Avg Wall τ_w (Pa):	Max V_{outlet} (m/s):	Avg $P_{dynamic@outlet}$ (Pa):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:	Pressure Loss $(Pa)_{static}$:	% Difference:
k-w standard$_{transitional}$	Uniform Velocity	2.159299	0.885612	0.014152	1.657279	0.301573	1.572234	1.645368	1.289912	27.556529
k-w standard$_{transitional}$	Periodic	1.124335	0.821048	0.011997				1.303506	1.289912	1.053816
k-w standard$_{transitional}$	Periodic Velocity Profile	1.173188	0.819754	0.011964	1.852350	0.310434	1.296149	1.295385	1.289912	0.424270
Laminar$_{fine\text{-}mesh}$	Uniform Velocity			0.014165	1.658291	0.302636	1.572386	1.641460	1.289912	27.253570
Laminar$_{fine\text{-}mesh}$	Periodic			0.012000				1.303089	1.289912	1.021481
Laminar$_{fine\text{-}mesh}$	Periodic Velocity Profile			0.011939	2.049223	0.311829	1.294567	1.295340	1.289912	0.420750

Flow Rate: ~ 230 slpm								Data Comparison		
								(Computational)	(Experimental)	(Gauge)
Viscous Model	Inlet BC:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Avg Wall τ_w (Pa):	Max V_{outlet} (m/s):	Avg $P_{dynamic@outlet}$ (Pa):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:	Pressure Loss $(Pa)_{static}$:	% Difference:
k-w standard$_{transitional}$	Uniform Velocity	1.830486	0.756706	0.010317	1.308848	0.171367	1.135779	1.178898	0.931647	26.539139
k-w standard$_{transitional}$	Periodic	0.980842	0.709082	0.008951				0.969848	0.931647	4.100348
k-w standard$_{transitional}$	Periodic Velocity Profile	0.992176	0.708196	0.008929	1.571950	0.175912	0.966847	0.966797	0.931647	3.772916
Laminar$_{fine\text{-}mesh}$	Uniform Velocity			0.010328	1.310052	0.171955	1.135566	1.176332	0.931647	26.263670
Laminar$_{fine\text{-}mesh}$	Periodic			0.008937				0.967307	0.931647	3.827663
Laminar$_{fine\text{-}mesh}$	Periodic Velocity Profile			0.008923	1.577651	0.175987	0.965702	0.965826	0.931647	3.668608

Flow Rate: ~ 150 slpm								Data Comparison		
								(Computational)	(Experimental)	(Gauge)
Viscous Model	Inlet BC:	Max Wall Y+ Value:	Avg Wall Y+ Value:	Avg Wall τ_w (Pa):	Max V_{outlet} (m/s):	Avg $P_{dynamic@outlet}$ (Pa):	Pressure Loss $(Pa)_{total}$:	Pressure Loss $(Pa)_{static}$:	Pressure Loss $(Pa)_{static}$:	% Difference:
k-w standard$_{transitional}$	Uniform Velocity	1.448741	0.608241	0.006654	0.933122	0.076993	0.725004	0.745348	0.597240	24.798844
k-w standard$_{transitional}$	Periodic	0.809779	0.577645	0.005942				0.642447	0.597240	7.569376
k-w standard$_{transitional}$	Periodic Velocity Profile								0.597240	
Laminar$_{fine\text{-}mesh}$	Uniform Velocity			0.006662	0.933574	0.077264	0.724914	0.743891	0.597240	24.554893
Laminar$_{fine\text{-}mesh}$	Periodic			0.005937				0.641773	0.597240	7.456546
Laminar$_{fine\text{-}mesh}$	Periodic Velocity Profile			0.005935	1.084493	0.078481	0.641338	0.641462	0.597240	7.404473

END OF RESULTS

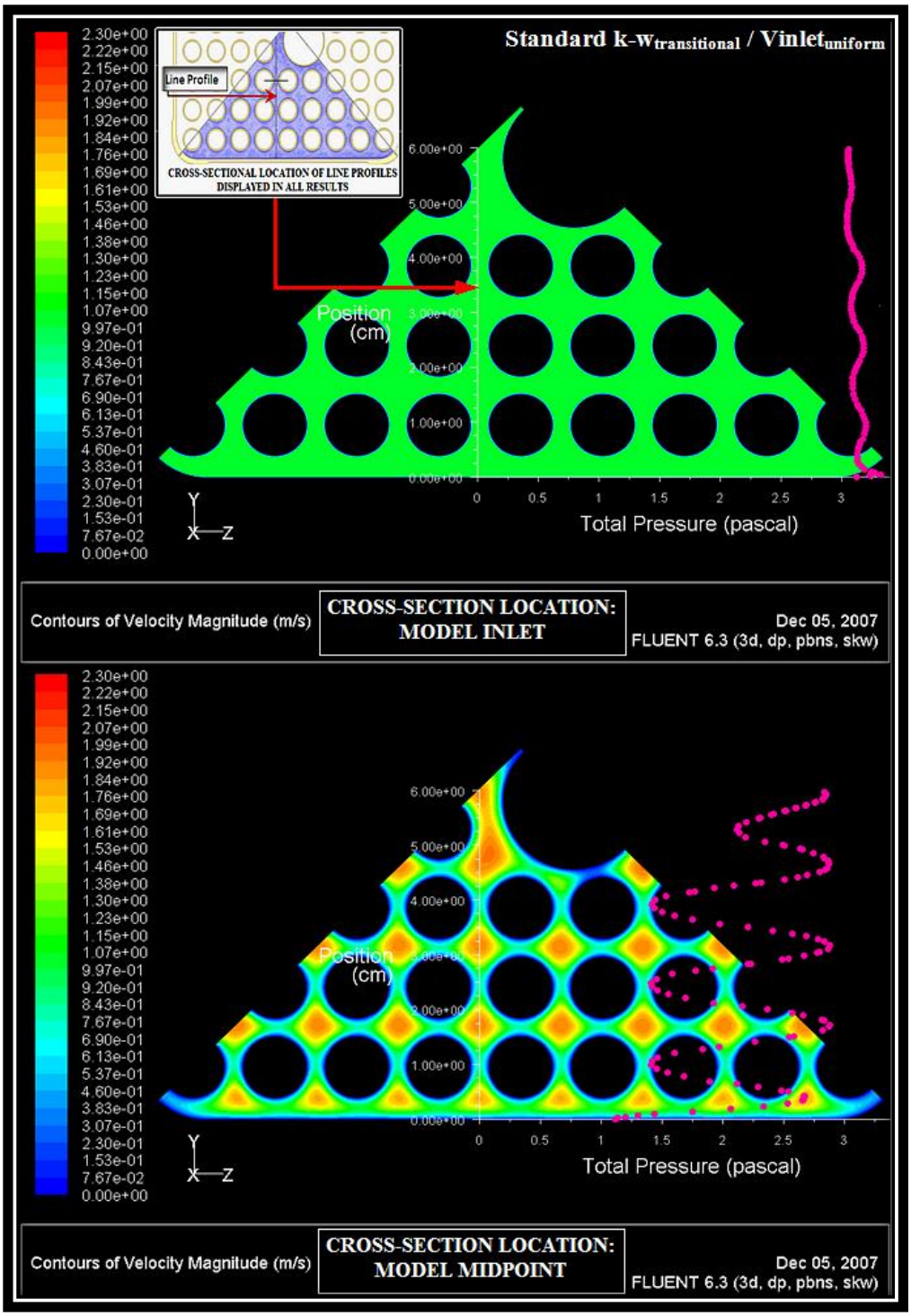


Figure 5.30: Contours of velocity magnitude at specified cross-sections with total pressure line profiles located along y-axis at approximately 450 slpm.

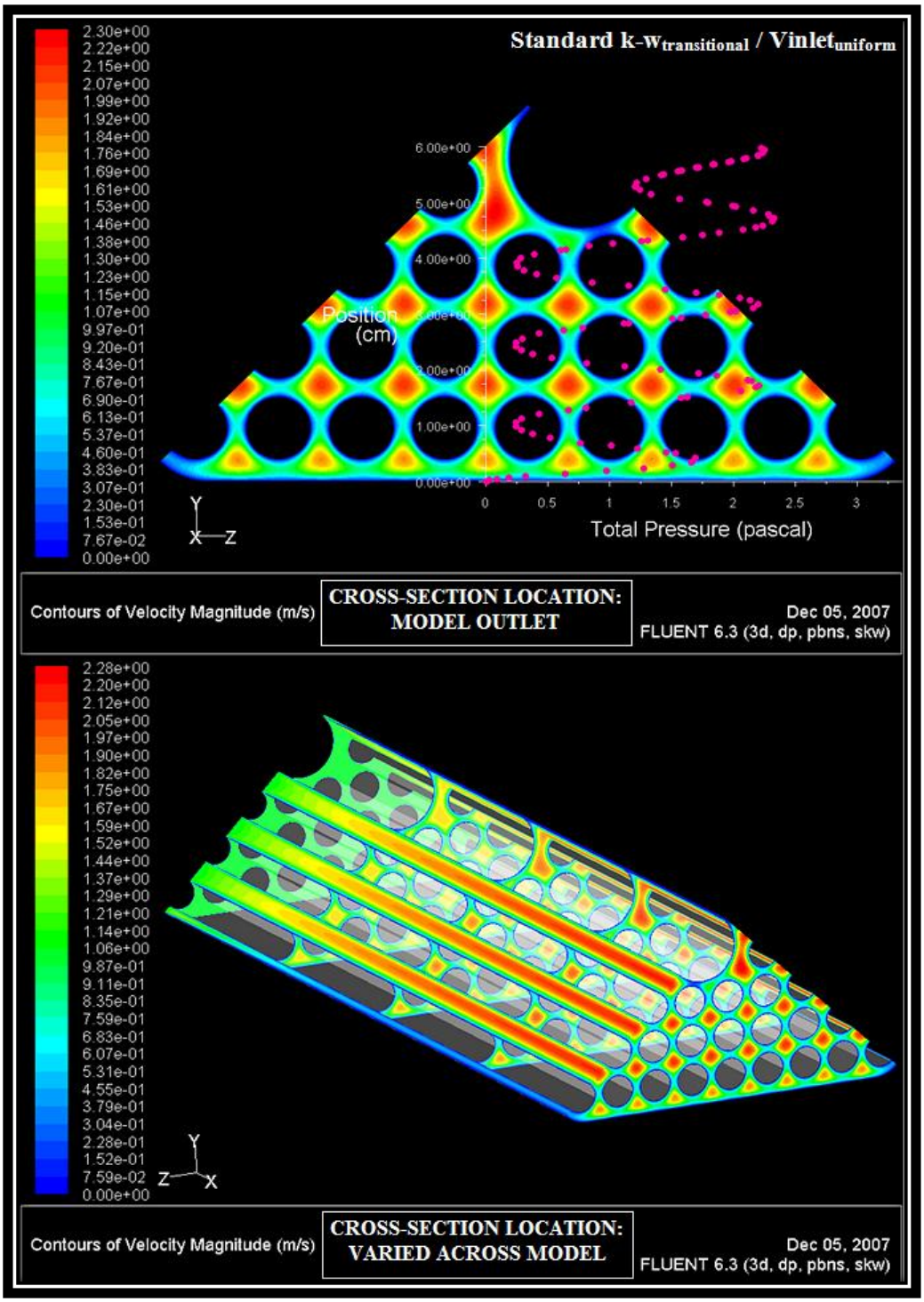


Figure 5.31: Contours of velocity magnitude at specified cross-sections with total pressure line profile located along y-axis at approximately 450 slpm.

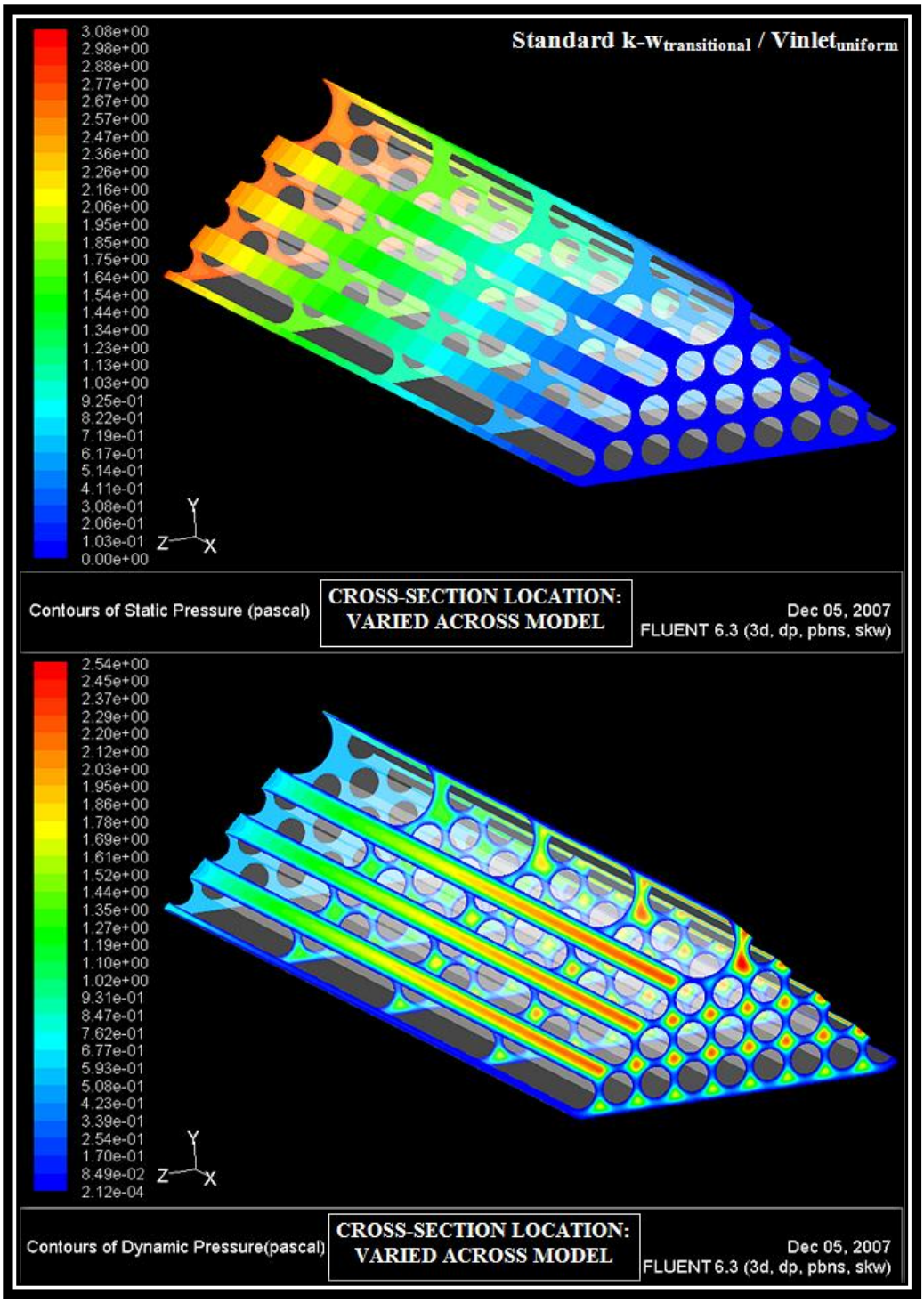


Figure 5.32: Contours of static and dynamic pressure across specified cross-sections at approximately 450 slpm.

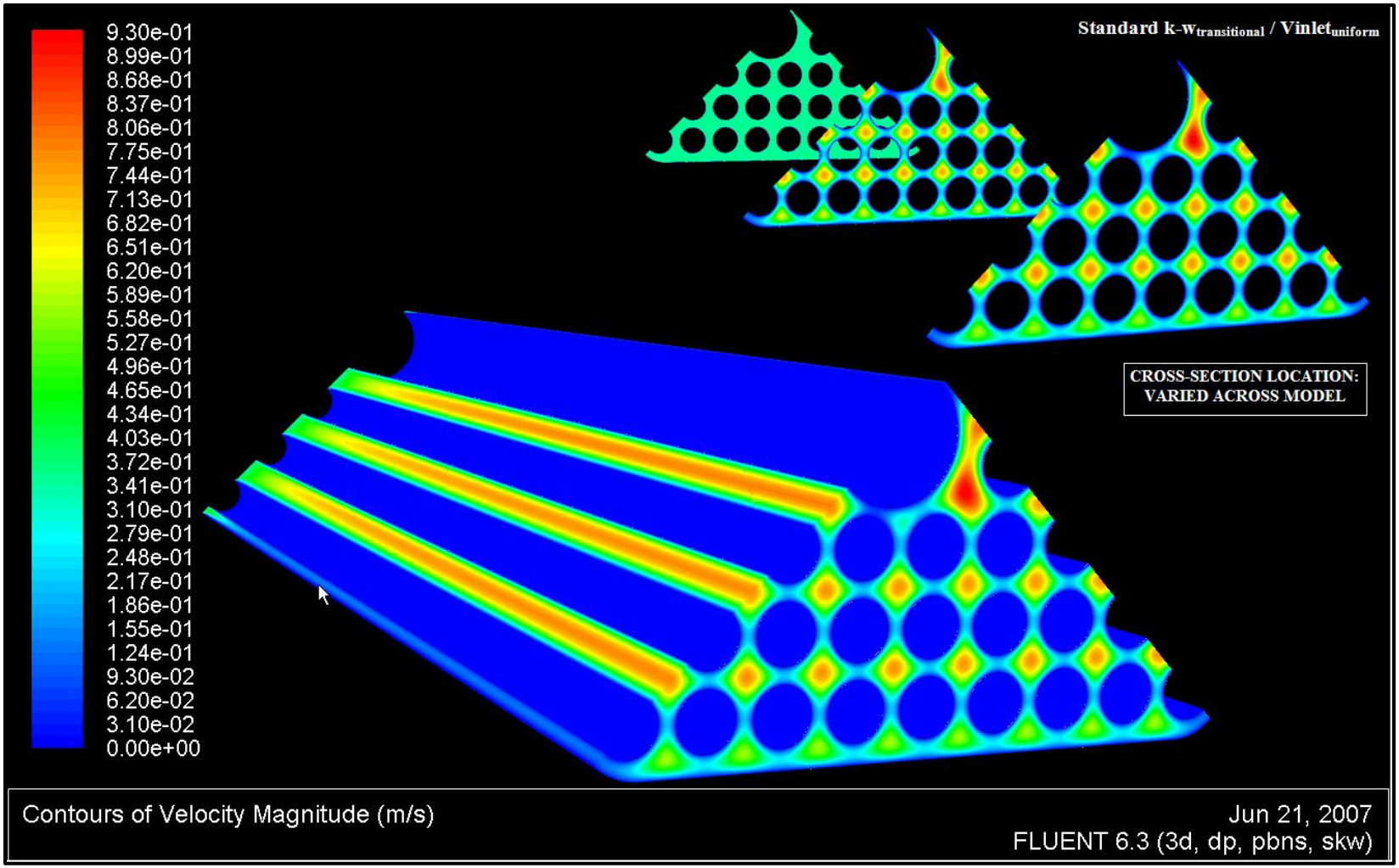


Figure 5.33: Contours of velocity magnitude at specified cross-sections and symmetry planes at approximately 150 slpm.

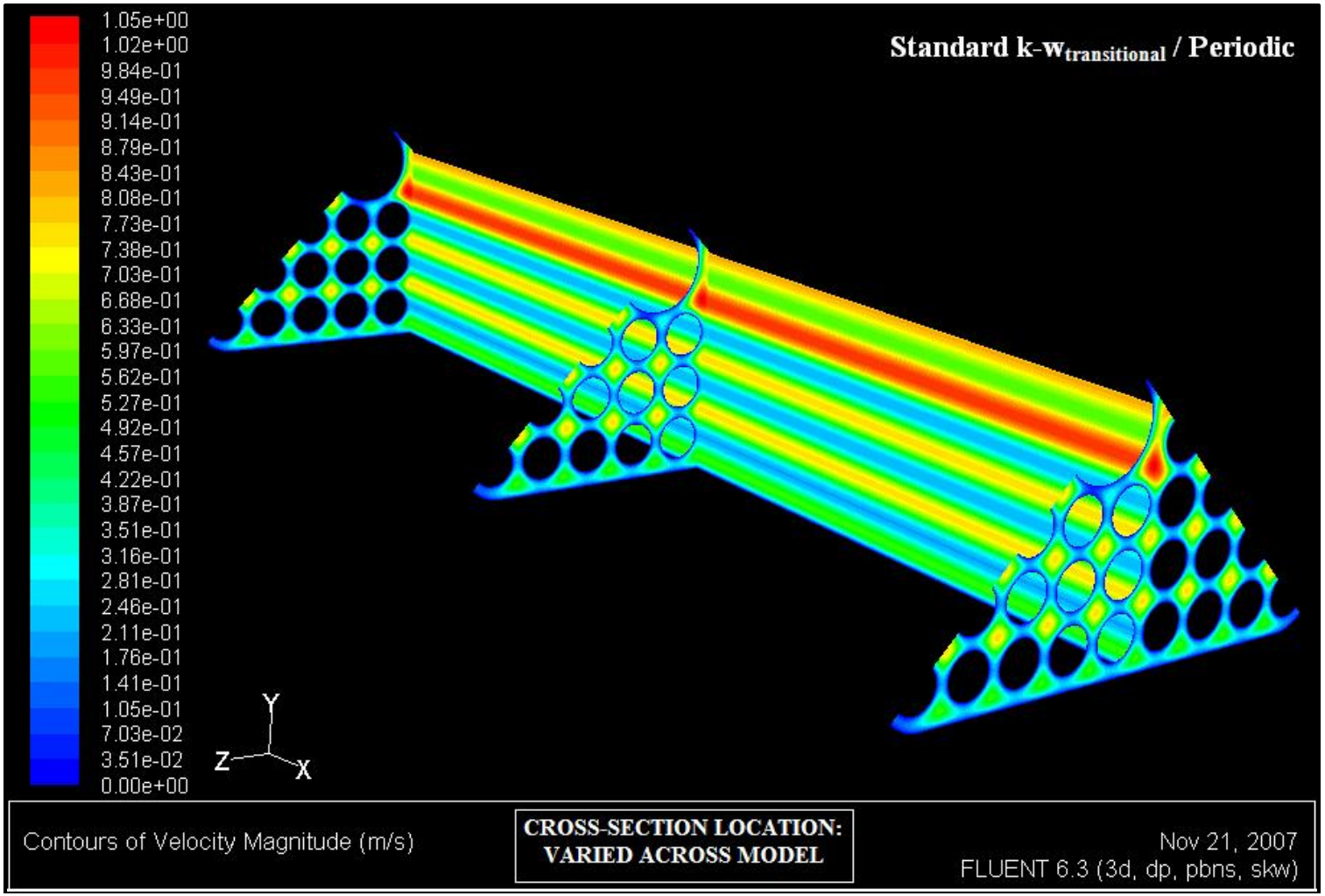


Figure 5.34: Contours of velocity magnitude at specified cross-sections and horizontal midplane at approximately 150 slpm.

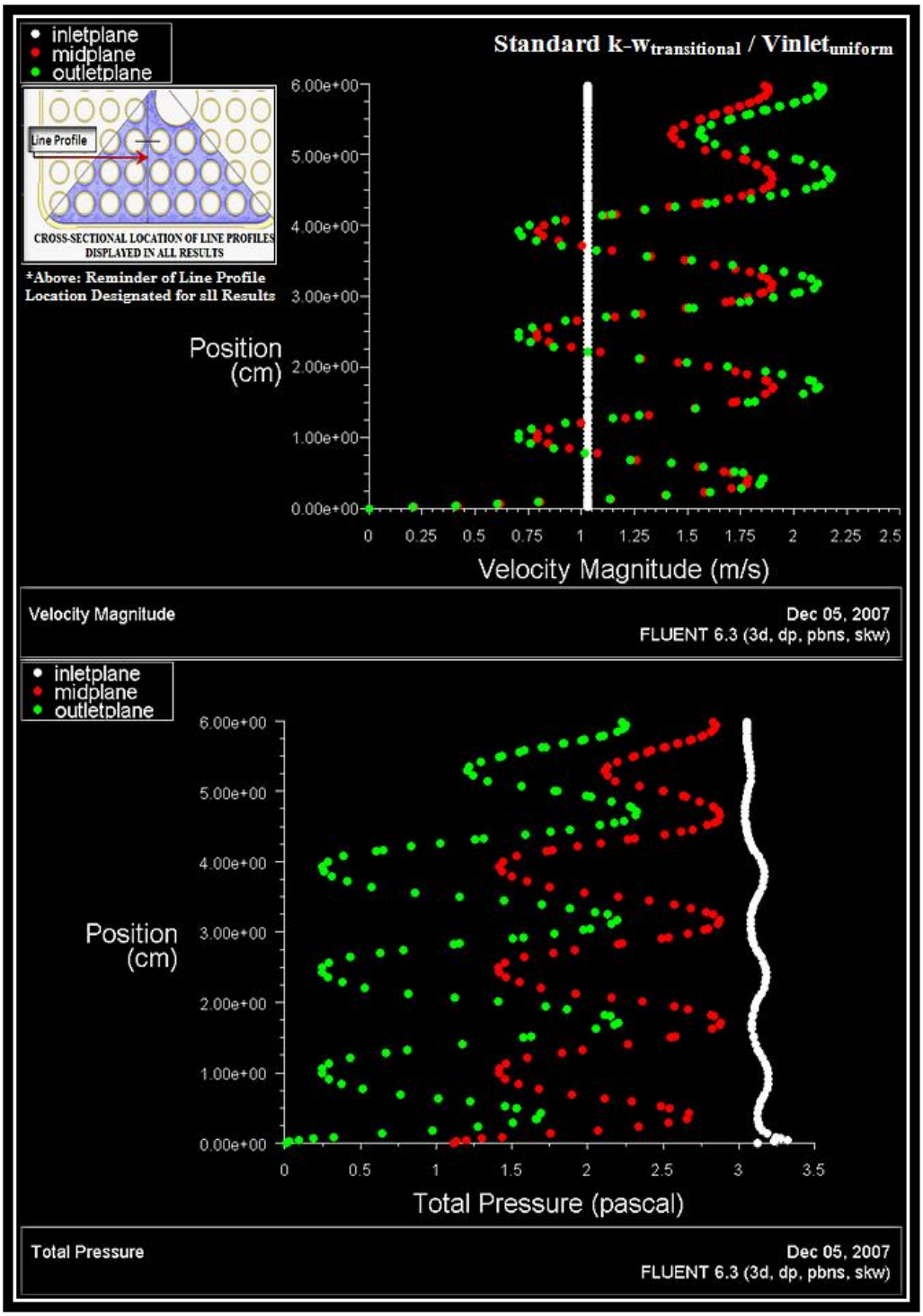


Figure 5.35: Velocity magnitude and total pressure line profiles along y-axis for cross-section planes specified at approximately 450 slpm.

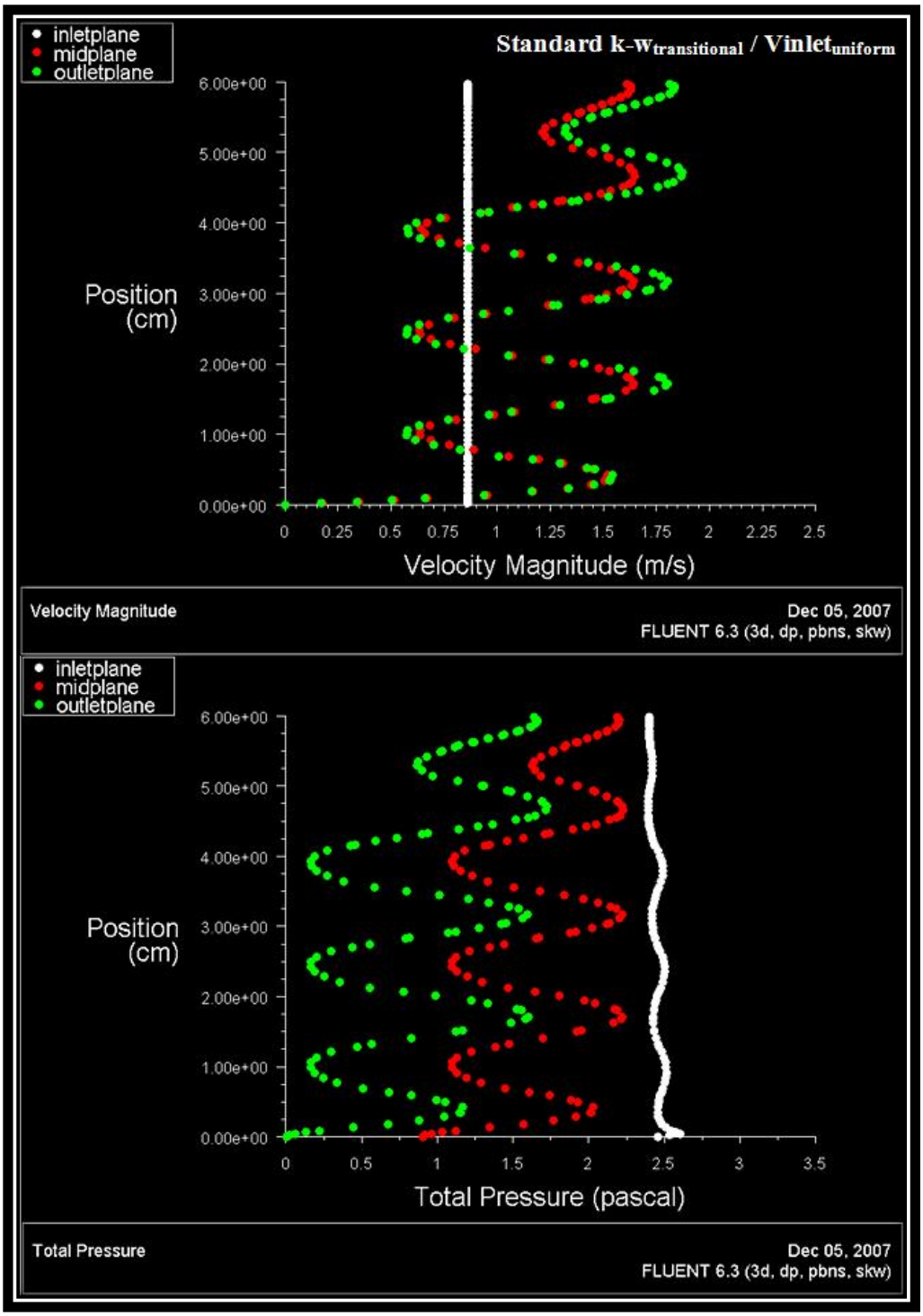


Figure 5.36: Velocity magnitude and total pressure line profiles along y-axis for cross-section planes specified at approximately 380 slpm.

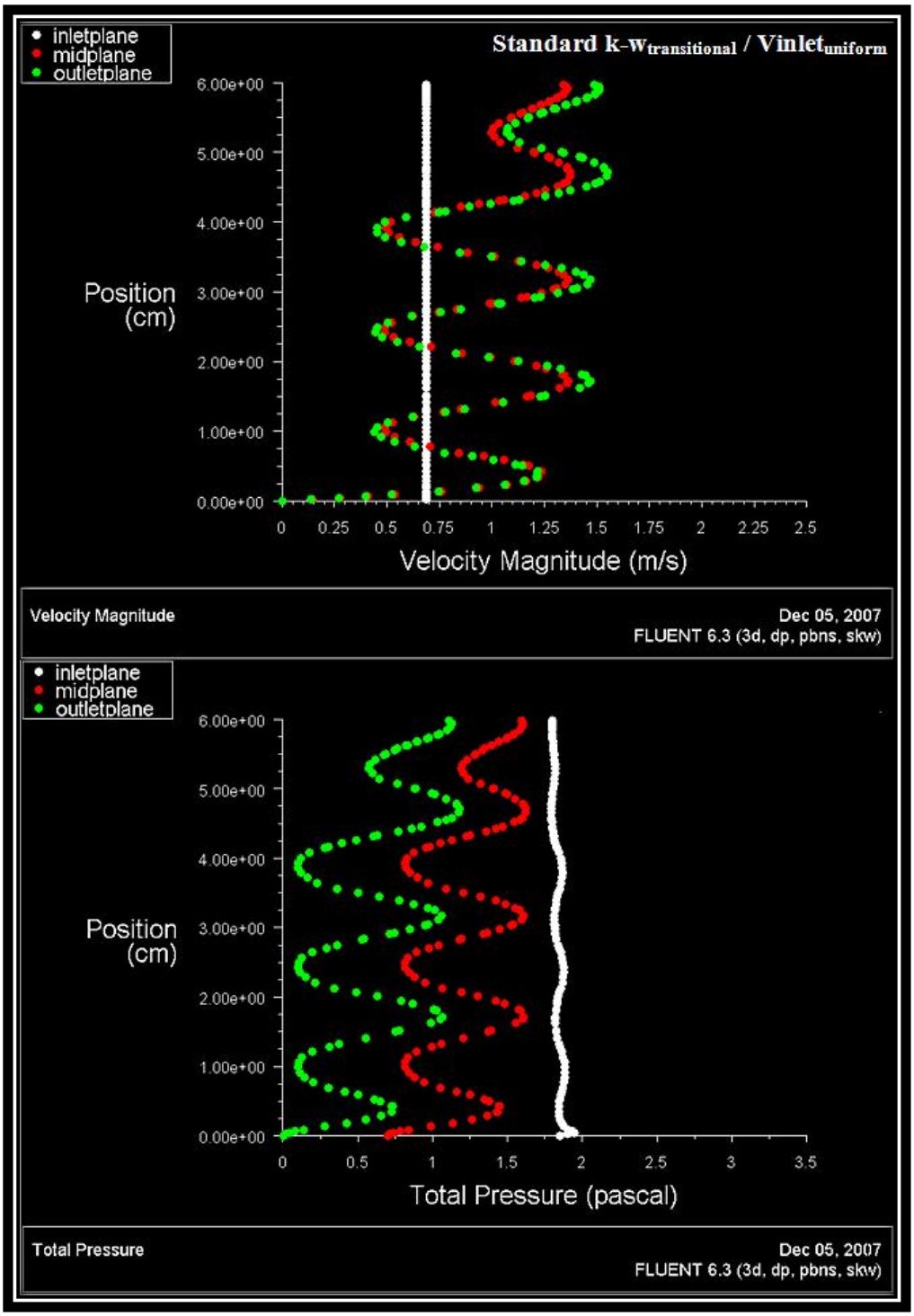


Figure 5.37: Velocity magnitude and total pressure line profiles along y-axis for cross-section planes specified at approximately 300 slpm.

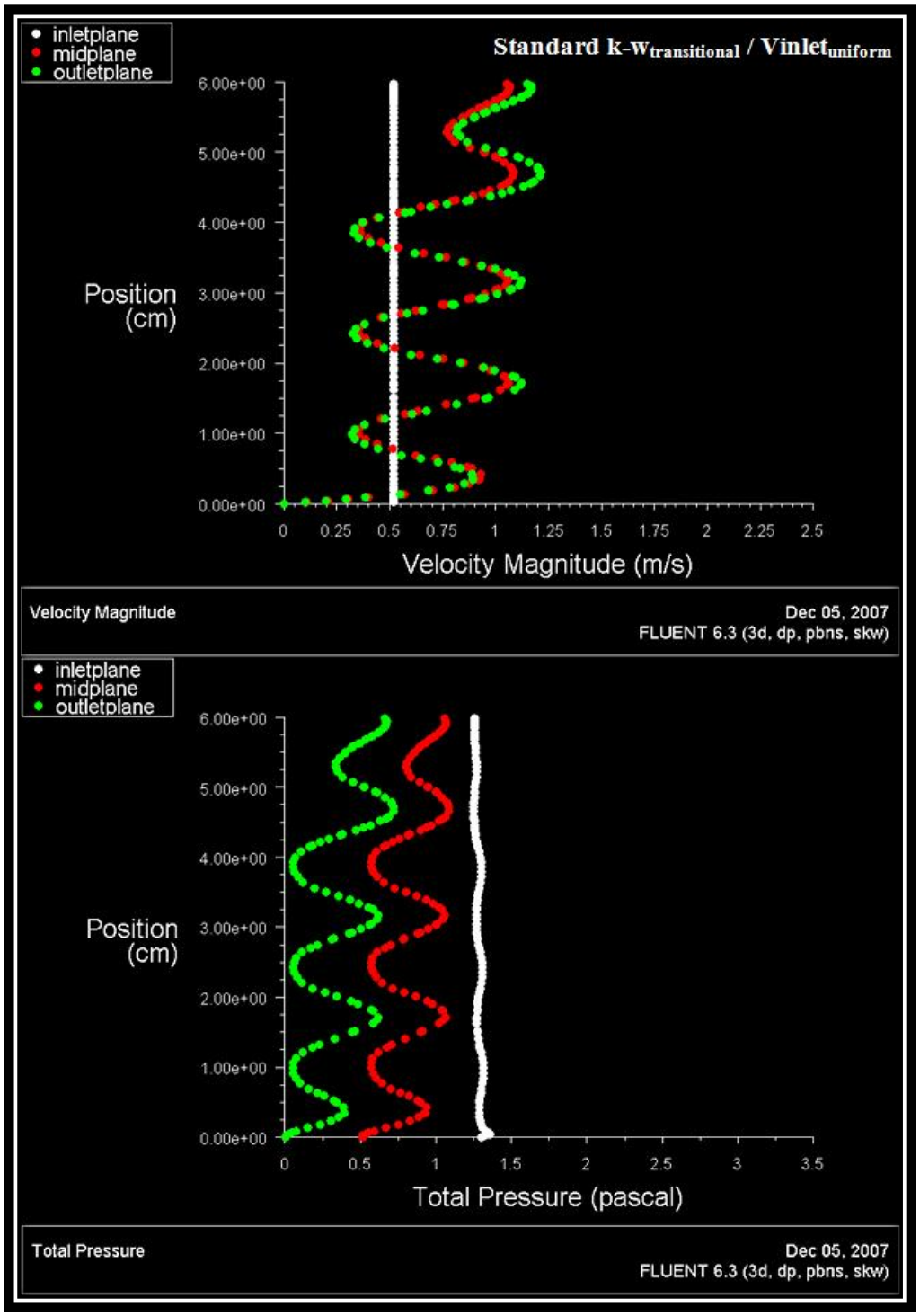


Figure 5.38: Velocity magnitude and total pressure line profiles along y-axis for cross-section planes specified at approximately 230 slpm.

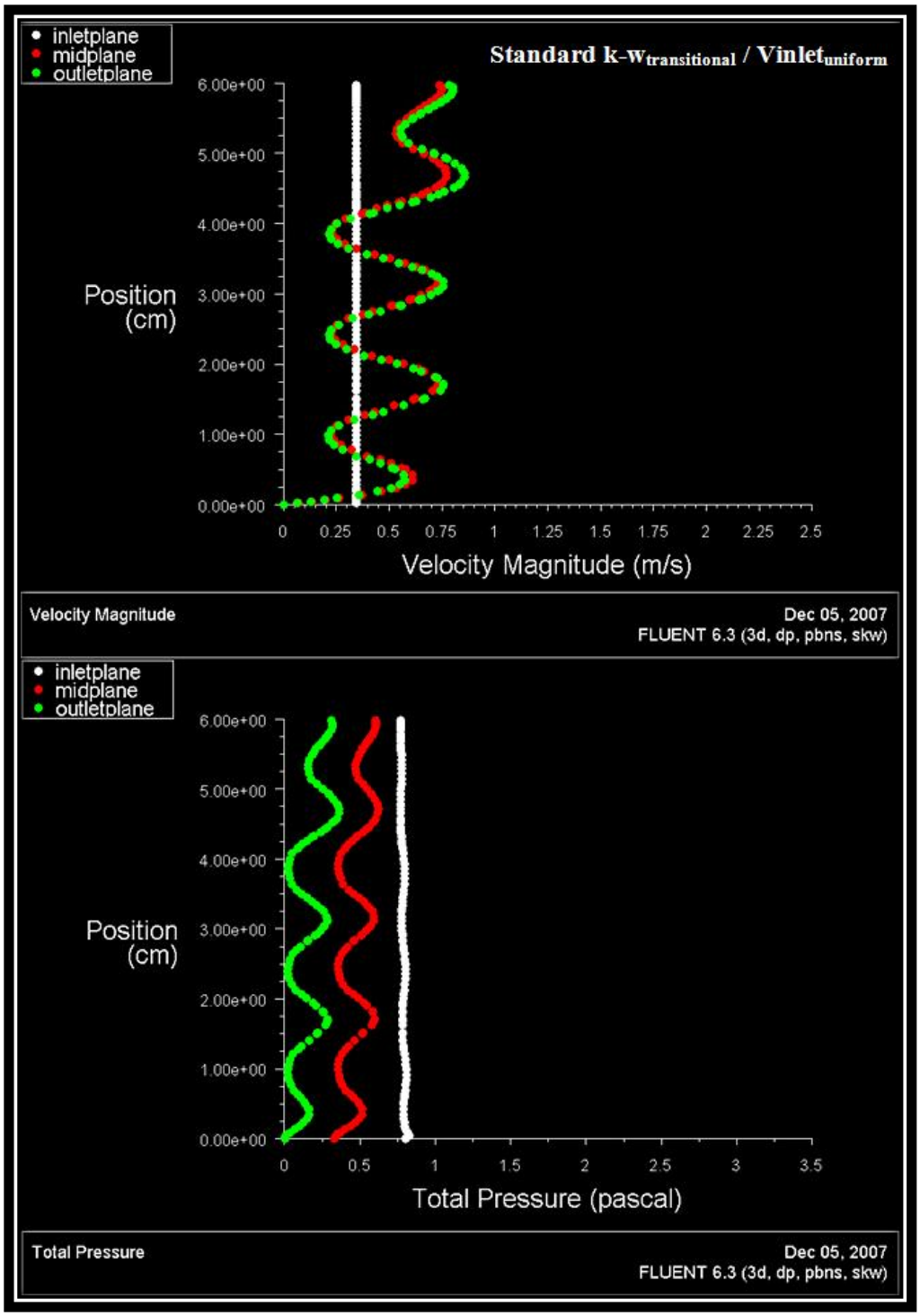


Figure 5.39: Velocity magnitude and total pressure line profiles along y-axis for cross-section planes specified at approximately 150 slpm.

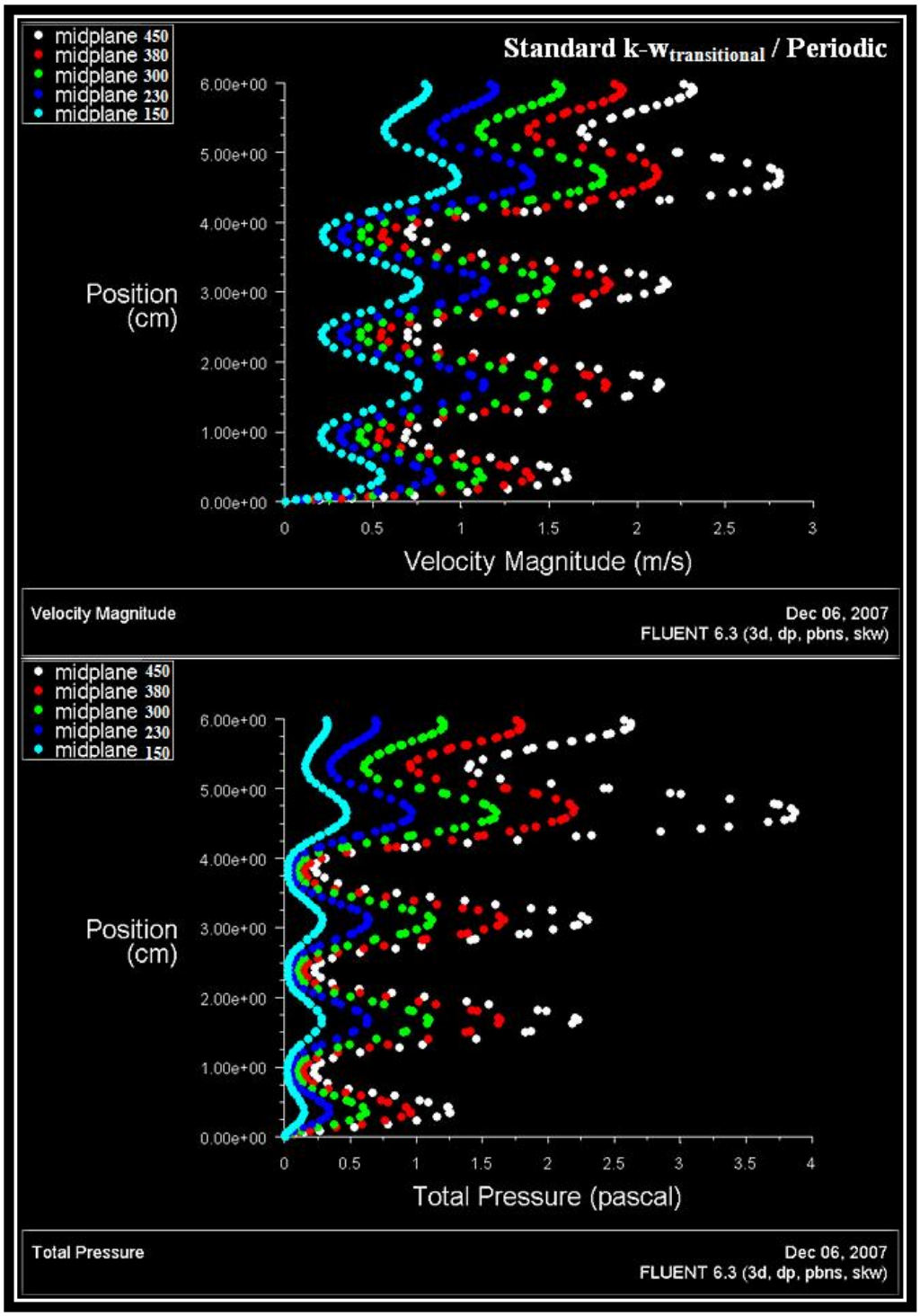


Figure 5.40: Velocity magnitude and total pressure line profiles along y-axis for cross-section plane specified at all airflow rates.

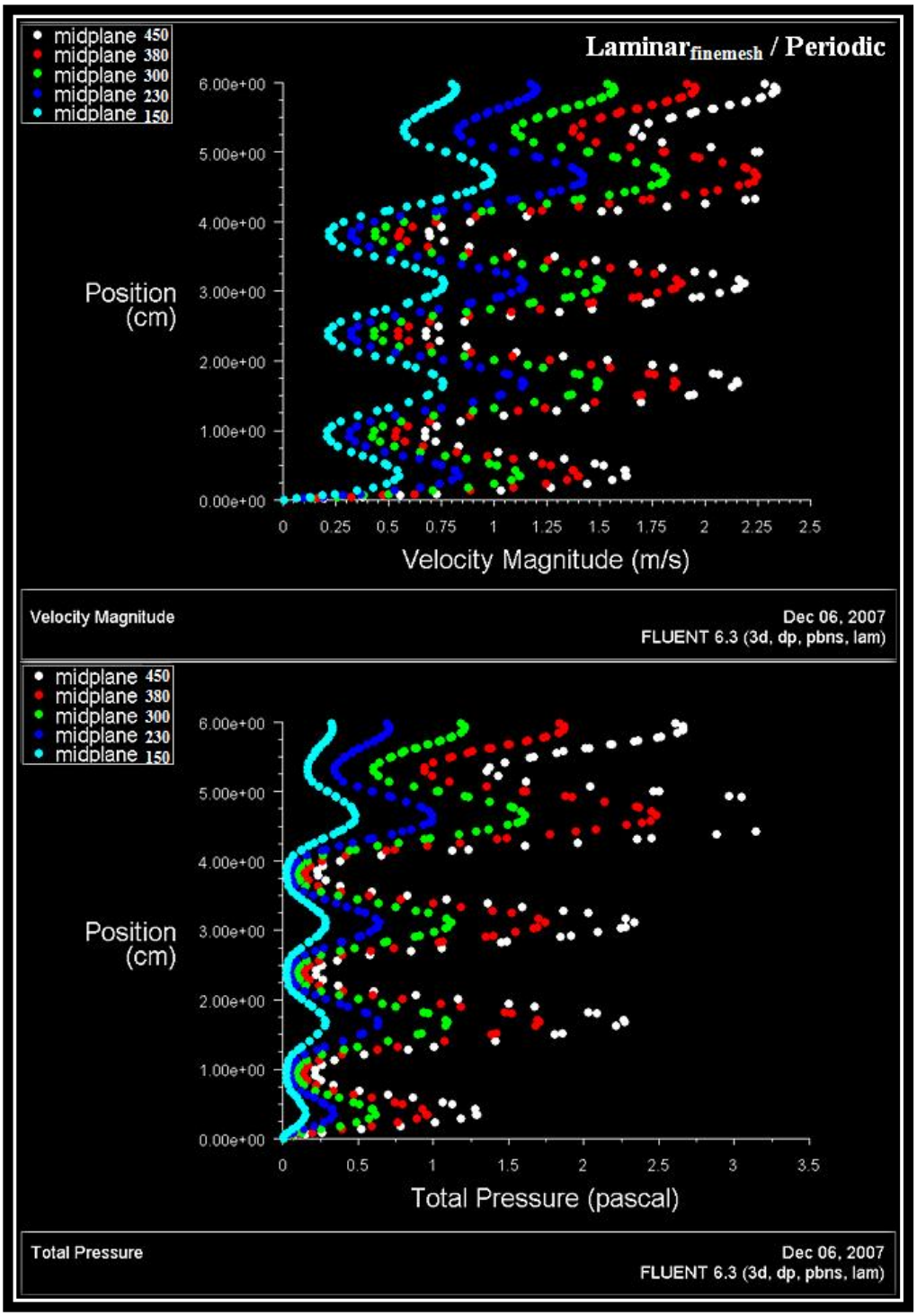


Figure 5.41: Velocity magnitude and total pressure line profiles along y-axis for cross-section plane specified at all airflow rates.

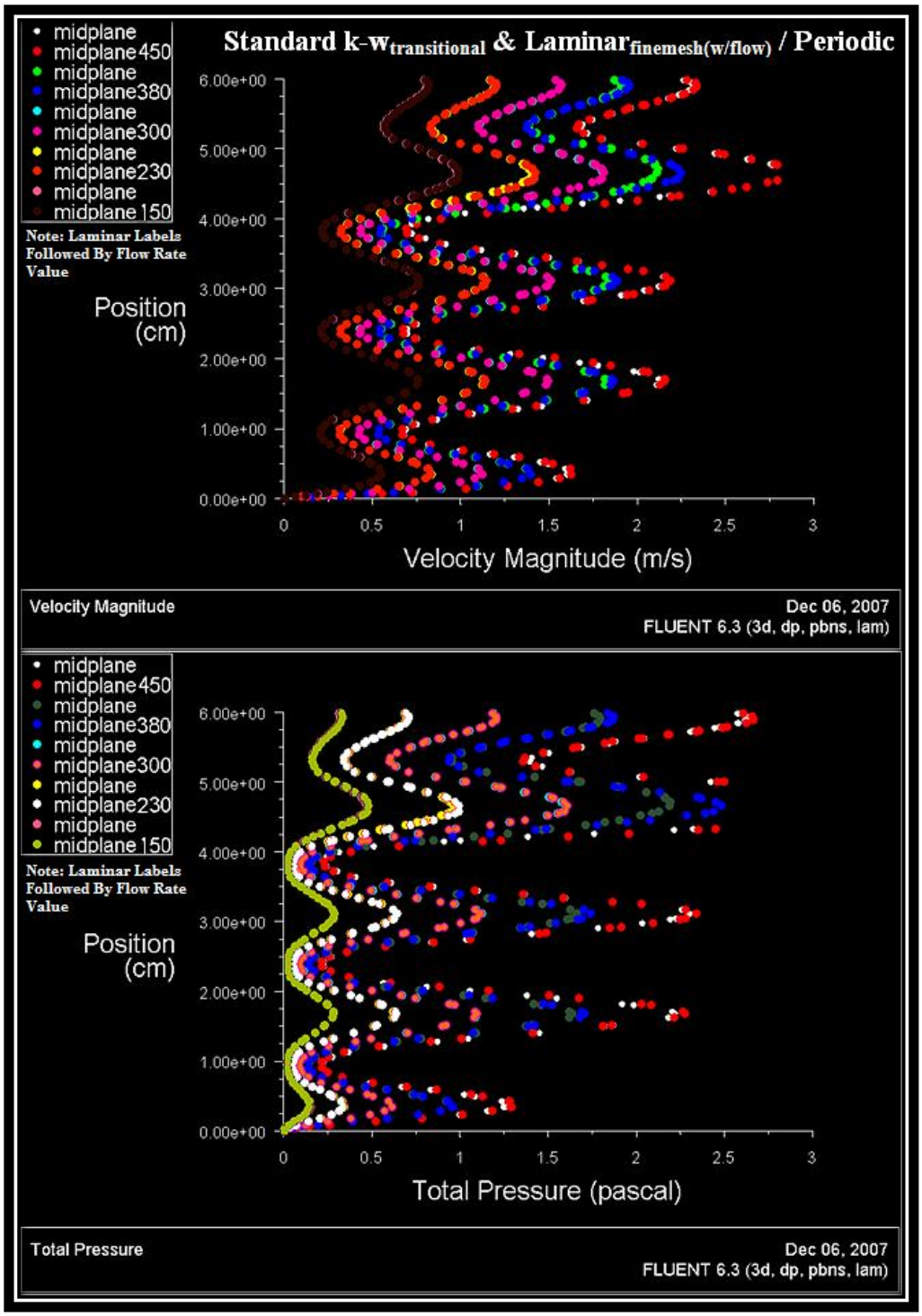


Figure 5.42: Velocity magnitude and total pressure line profiles compared from both viscous models at y-axis for cross-section plane specified at all airflow rates.

In-depth postprocessing of detailed two- and three-dimensional graphical representations for airflow developments in each of the numerous computational trials conducted is a clearly impracticable undertaking in this situation. However, the combined provision of numerically tabulated parameter results and less elaborate depictions of multiple flow-field solutions in conjunction with a modest postprocessing effort for high-end graphical images of central influence to the flow development analysis, is sufficiently adequate for thorough examination of the developing flow field under varying conditions imposed and quantification of influences on resolved solutions. Jointly considered with the preliminary model assessments above, the overall objective to gain an acceptable understanding of considerations pertaining to implementation of a suitable computational model including: mesh resolution, type and form of boundary conditions applied, viscous flow treatment options, convergence criteria, and computational resource requirements is largely achieved.

Although a direct comparison between these preliminary model solutions and SNL experimental measurements is not truly possible without consideration of interstitial spacer components along the fuel bundle domain, these results provide substantial evidence that actual flow fields traversing bundle segments within the experimental fuel assembly are well-represented by the FLUENT CFD models. As a result, a few of the most pertinent results and observations arising from initial flow-field examinations of the fuel bundle run model (lending toward suppositions of particular interest), are briefly outlined for discussion below.

Significant Results and Initial Observations:

- Foremost, despite a number of minimal discrepancies between computational and experimental trials, a remarkable agreement is observed for the predicted pressure

drops across the fuel bundle assembly in all flow rate cases between measurements represented by the hydraulic analysis curve-fit relationships from SNL (with derived loss coefficients) and the standard k-ω transitional viscous model in FLUENT with periodic boundaries implemented. The solution values for characteristic flow parameters predicted by the fuel bundle model, including comparative differences in the recorded pressure losses with SNL experimental results, are presented in Tables 5.6 and 5.7 above. This numerical data suggests an average pressure loss difference of less than 2% and a maximum of approximately 7.6%, which corresponds to the lowest flow rate.

- At first glance, this appears contradictory to the expectation of a lower percent difference between experimental and computational values for the lower airflow rates. However, a substantial fraction of this difference is attributed to approaching limitations of physical measurement accuracy considering the extremely small differential pressures measured across the domain (less than 1 Pa) and the systematic influences arising from a slight incongruence between research models utilized. Nonetheless, numerical values for all five airflow rates involving some type of flow periodicity are well within acceptable limits and actually exceed initial expectations.
- Concerns surrounding the lack of an inlet velocity profile and the effect a designated inlet profile incurs on the solution obtained are emphasized by the large differences in the predicted pressure losses for trials incorporating either the uniform or fully-developed flow profile assumption. On the other hand, the closeness of results with periodic flow profiles to the SNL experiments serves as a critical degree of merit for substantiating the use of a corresponding, repeated flow profile representing near-periodic development for subsequent larger-scale

model analyses. This is specifically relevant to modeled flow domains of repeated, single-length spans comprising a fully-populated bundle run and spacer component as the individual flow segment targeted for evaluation.

- Furthermore, assuming the SNL experimental measurements are indeed accurate, the minute differences with the simulated pressure drops under cases of fully-developed, streamwise-periodic flow infers several characterizations with regard to the flow development at distances of approximately 10 to 15 hydraulic diameters removed from an upstream spacer component. Namely, the effects of viscous shear forces and increased static pressure on the fluid as it rapidly expands on exit from the spacer to refill the fuel bundle volume are largely minimized through dissipation of fluid energy at this point by established redevelopment of the majority flow field.
- Hence, the supposition suggesting that a repeating flow-field development is quickly re-established and velocity gradients due to acceleration/deceleration in the streamwise direction are sizably reduced in terms of significant pressure loss is afforded some level of supporting evidence, which permits plausible justification for assuming a constant linear pressure loss over most of the bundle run length.
- Likewise, the well-known constitutive/phenomenological equations used by SNL for correlating experimental curve fits to the recorded pressure drop data and inherent assumptions limiting their application to certain conditions are similarly provided with a measure of assurance in regard to a correct form of implementation in the final hydraulic analysis report (at least to the preliminary extent of these assessments).

- Further investigation into the applicability of a laminar viscous model to resolve flow fields through a fuel bundle run at related airflow rates reveals a surprisingly real possibility for accurate employment within the current research, provided an equivalently refined mesh of turbulence model requirements for enhanced wall treatment is exploited in the computational trials. This is evident through comparison of laminar and standard k-ω transitional viscous model results for predicted pressure loss in Tables 5.6 and 5.7 above.
- On the contrary, some discrepancies in the maximum velocity reported at the flow outlet exist between the laminar and standard k-ω model solutions with a periodic velocity profile applied at the inlet boundary. This indicates the potential misrepresentation of the velocity flow-field profile near the outlet boundary by one of these viscous models, to at least a minimal extent over the cross-sectional flow plane. Although such observations generally warrant additional examination, a current assumption is accepted that these velocity profile differences are of negligible concern due to the overall pressure loss agreement between solutions; but the issue is certainly open to revisiting at later stages of research development.
- As previously mentioned, simulation trials in which the converged velocity profile is captured from solutions of assumed flow periodicity and incorporated as a periodic velocity profile for the inlet boundary condition accompanied with ambient pressure outlet, are conducted in order to alleviate convergence concerns associated with the enforcement of periodic boundaries. Theoretically, adequate convergence of flow periodicity in the original trial is validated by applying the velocity profile solution as an inlet boundary condition in the second computation under above-specified conditions in which near-equivalent solutions for the

pressure loss are anticipated. This congruency among acquired solutions is observed within reasonable limits of accuracy for the predicted pressure losses of these numerical simulations as demonstrated in the results provided.

- A final observation of noteworthy importance is indicated by the establishment of velocity (as well as dynamic pressure) gradient patterns at all airflow rates with increasing flow development in the cross-sectional plane normal to fluid flow, but particularly with regard to the grouping of interstitial spaces between fuel rod rows from the canister wall to assembly center. Originally, based on decreasing wall contact surface per interstitial flow area in spaces adjacent to rows of fuel rod elements, a general trend of universally increasing average velocity is expected in rings of interstitial spaces from the canister wall (with highest contact to flow area ratio) inward to the assembly center. However, as unexpectedly shown in the graphical results above, the actual velocity trend resembles a more ripple-like effect in which the average velocity per interstitial row of spaces alternates in respective magnitude from the canister wall inward. Moreover, this effect is increasingly pronounced at increased levels of flow development along the fuel bundle length.
- For further clarification, these characteristics are captured by subsequent consideration of interstitial spaces in groups of successive radial rings from the inner canister wall to fuel bundle center through emphasis of a flow-field solution resulting from an aforementioned simulation trial and illustrated in Figure 5.43 below. This example corresponds to the computational trial implementing a uniform velocity inlet profile with standard k-ω transitional viscous model at approximately 150 slpm. The interstitial rings are evaluated by similarly located velocity line profiles as specified for all previous line profile results, at cross-

sectional planes of equally spaced intervals from inlet to outlet boundaries (*i.e.* cx_1 to cx_5) over the model length, but normalized by the maximum velocity value. Such characterizations of the developing flow field are of particular interest to future modeling endeavors and additional exploration of this subject is certainly recommended for subsequent research initiatives.

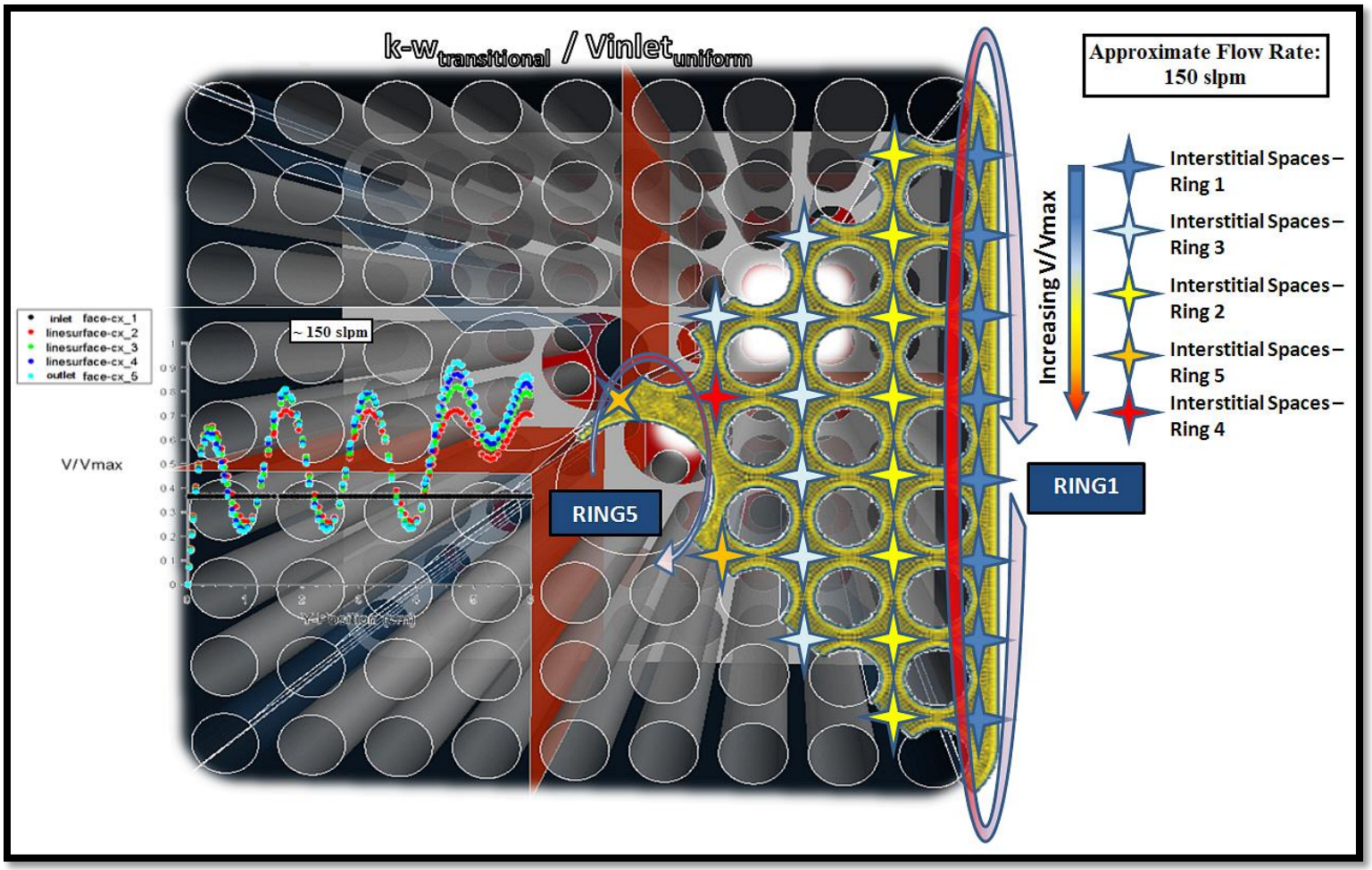


Figure 5.43: Example illustration of velocity profile distributions at several cross-sections along model length characterized by specific ring intervals for interstitial spaces among fuel rod rows using solution obtained from a selected trial run.

A substantial benefit afforded by commercial CFD software that is commonly referenced among the general CFD community is the capability of providing postprocessing for high-end graphics of resolved flow-field solutions. Due to the appeal of imparting such detailed graphics representing results for significant parameter values throughout the fuel assembly domain of interest to this research, a modest effort is expended in order to assess a greater extent of the postprocessing features available within the FLUENT CFD software. In particular, the ability to create three-dimensional overlay scenes for parameter values using various forms of representation on multiple surfaces is explored including direct consideration for resource costs associated with the production of these high-end graphics.

A few of the postprocessed depictions resulting from this effort are provided for reference in Figures 5.44 through 5.48 below, which emphasize the characteristics of flow development in selected computational trials implementing a uniform velocity inlet boundary condition and standard k-ω transitional viscous model at airflow rates of approximately 450 and 150 slpm. As expected, the concluding relationship observed for incurred costs (allocated time and computational requirements) is proportional to the degree of postprocessed graphical detail, complexity/size of the flow domain represented, and availability of high performance computing.

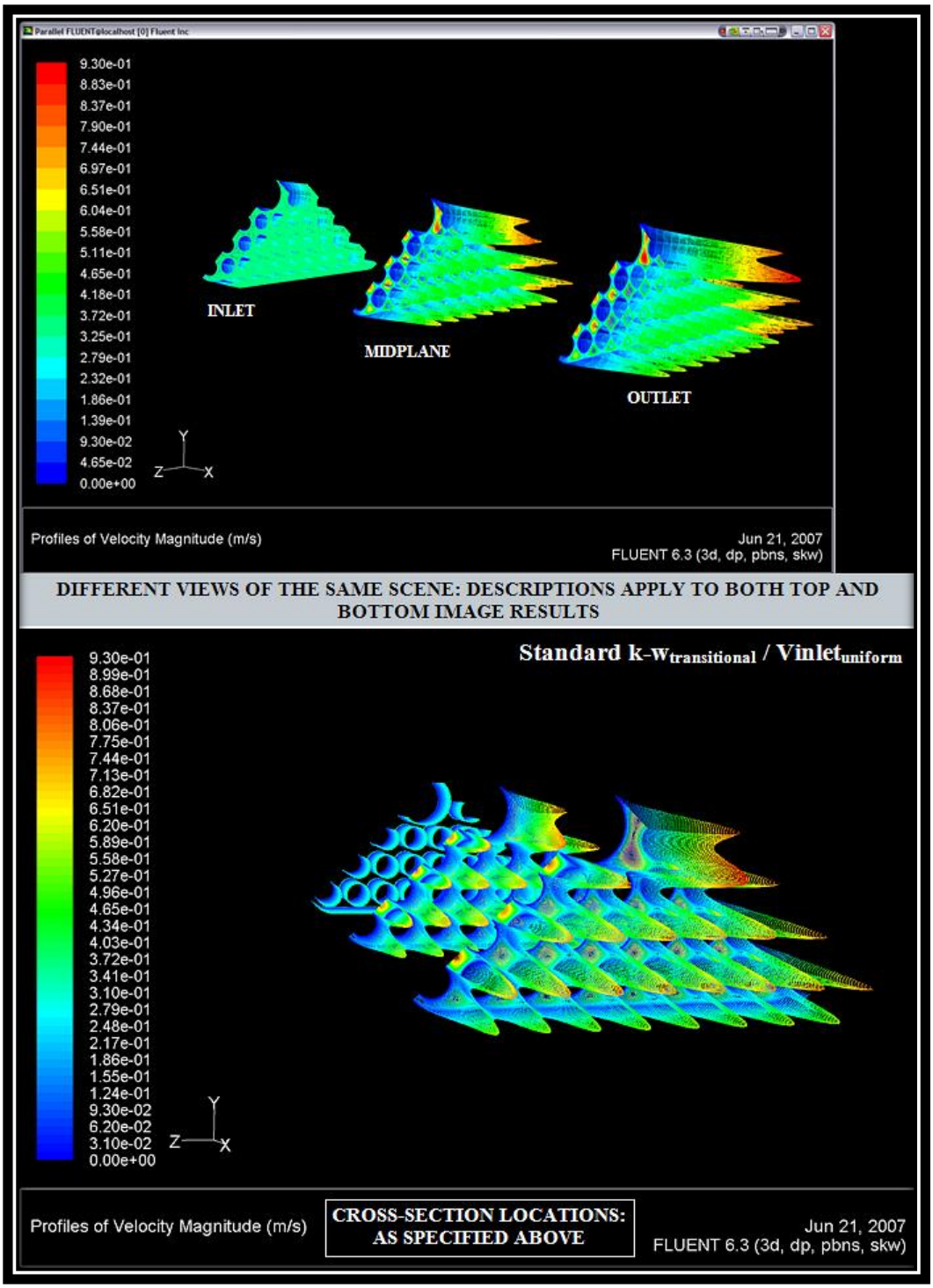


Figure 5.44: Contours of velocity magnitude with extruded 3D profiles overlaid at specified cross-sections for an airflow rate of approximately 150 slpm.

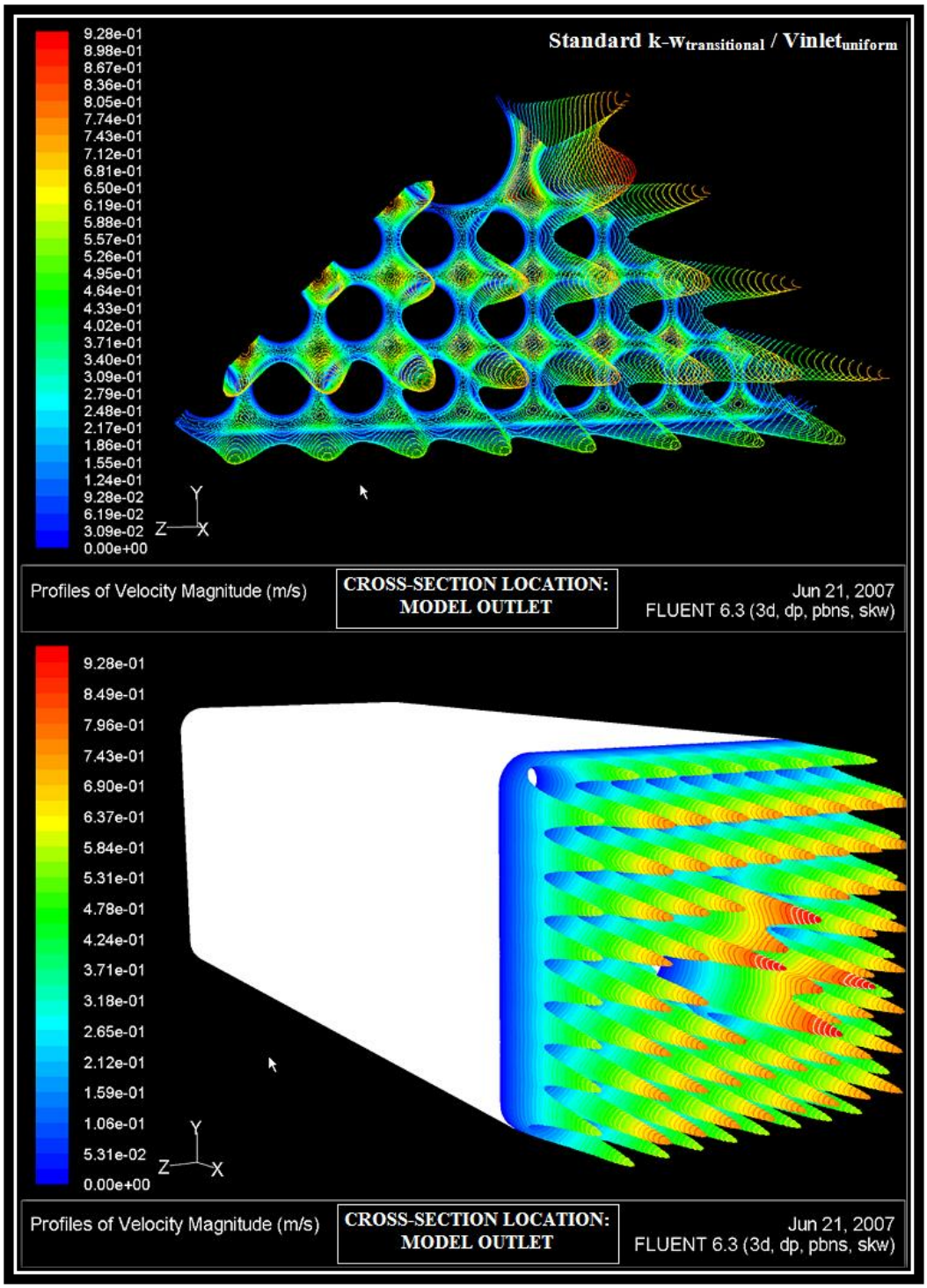


Figure 5.45: Unfilled (symmetry) and filled (full plane section) extruded 3D profiles of velocity magnitude at outlet for an airflow rate of approximately 150 slpm.

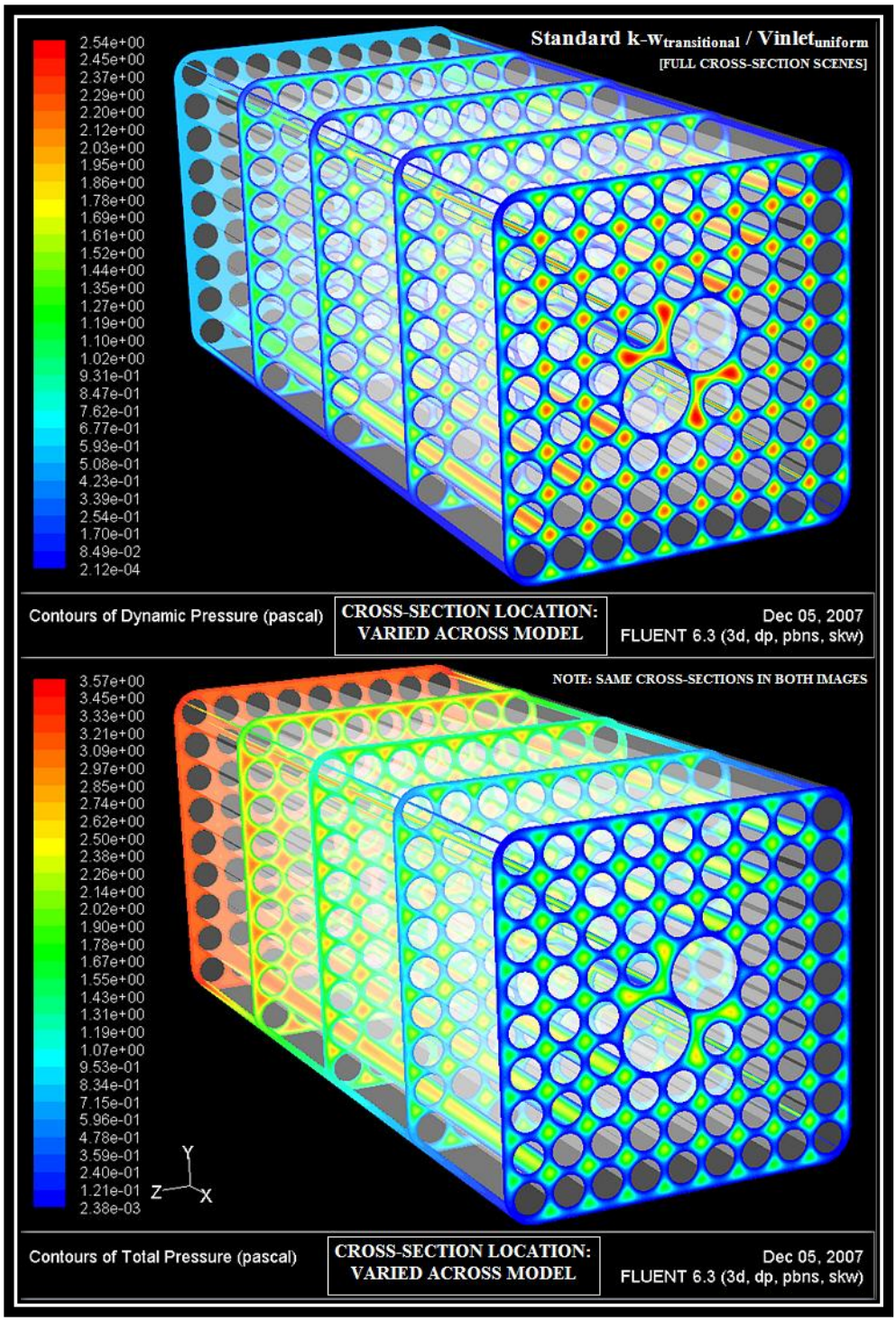


Figure 5.46: Full cross-section scenes exhibiting contours of dynamic and total pressure at specified plane sections for an airflow rate of approximately 450 slpm.

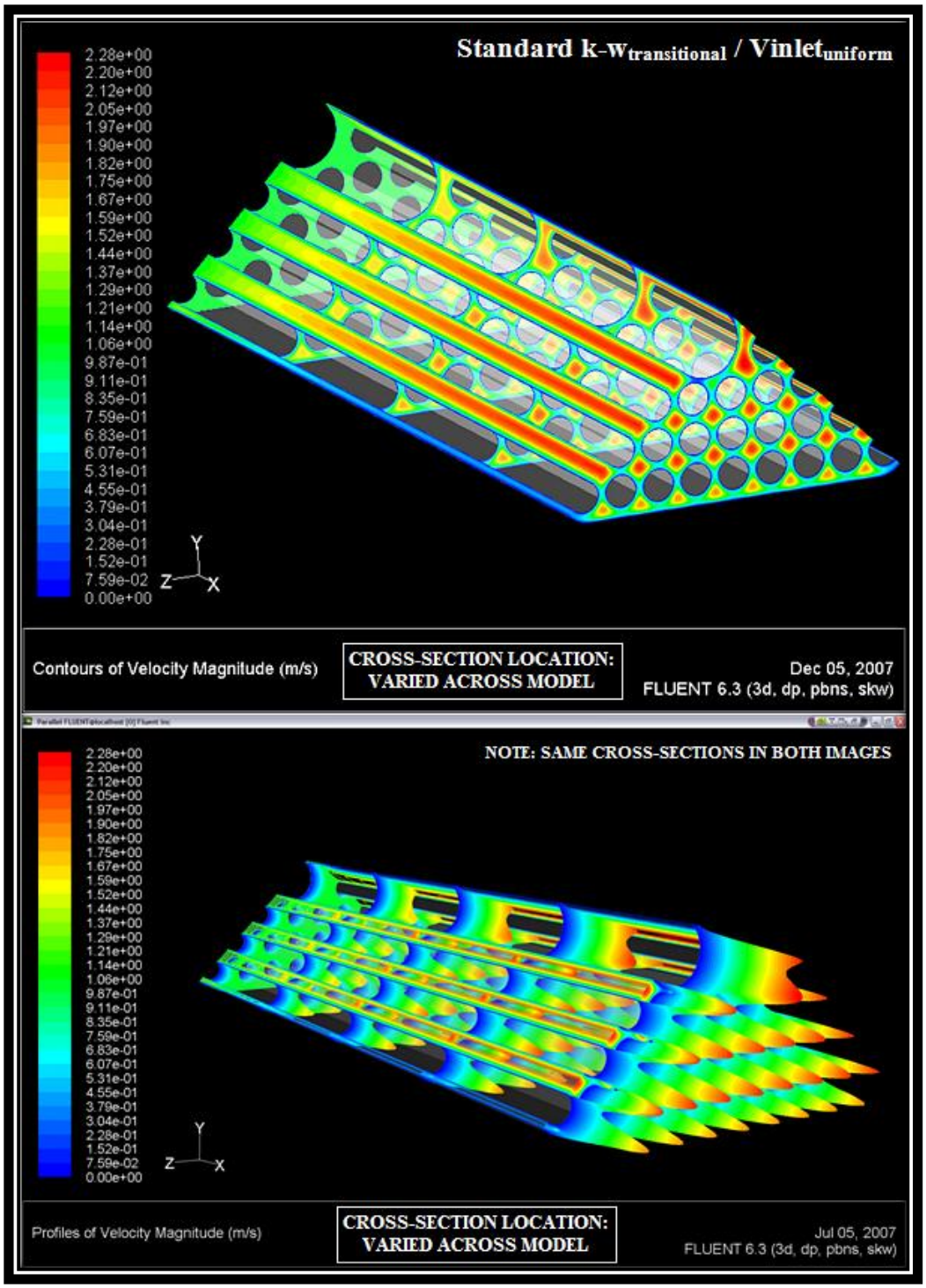


Figure 5.47: Contours of velocity magnitude with filled 3D profile equivalent representation for specified cross-sections at approximately 450 slpm.

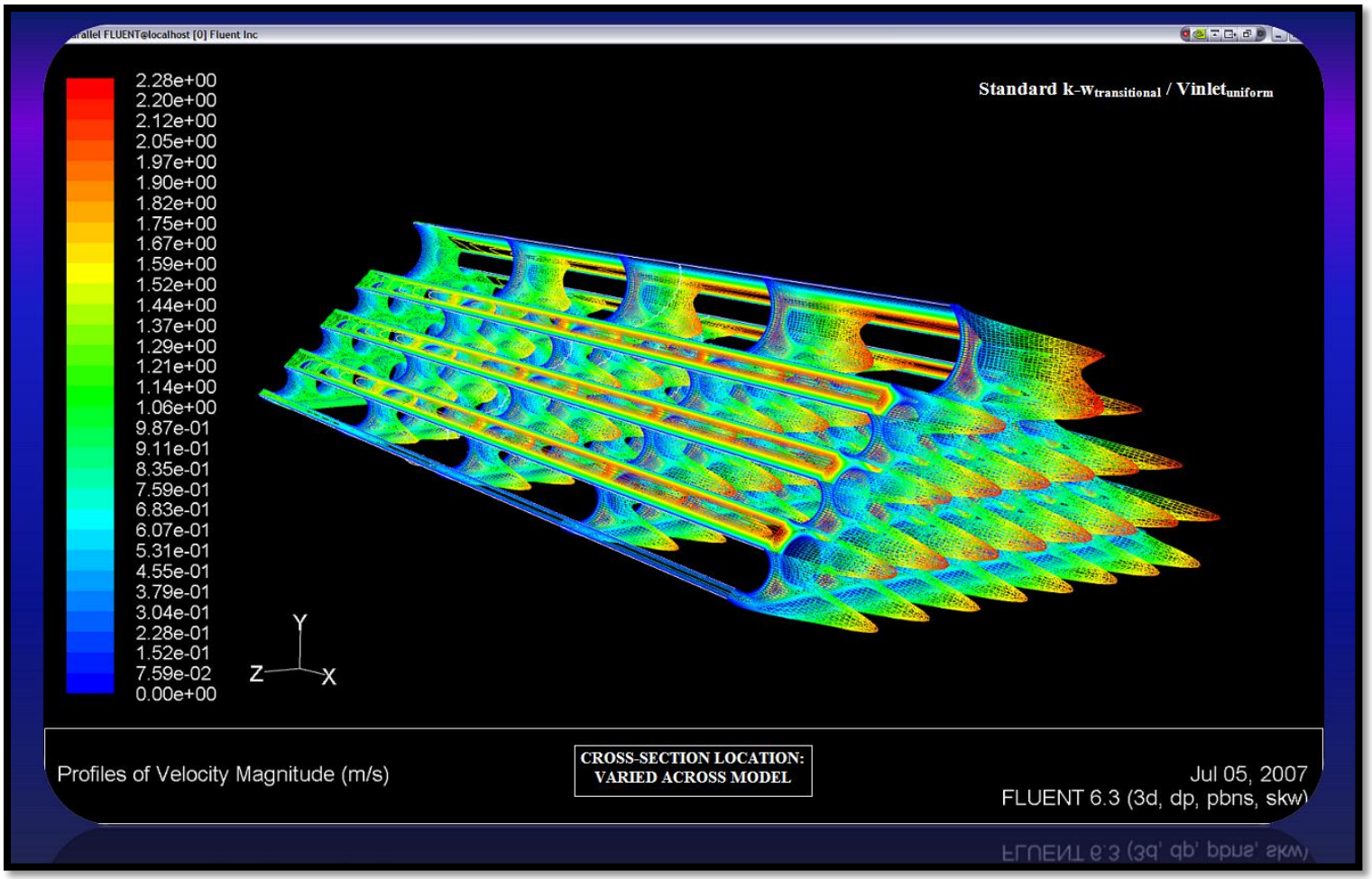


Figure 5.48: Highly detailed contours of velocity magnitude and overlaid extruded 3D profiles at specified cross-sections using transparent, unfilled mesh graphics for a designated airflow rate of approximately 450 slpm.

In conclusion, these extensive computational trials function as a phase of preliminary scoping studies for the assessment of fundamental considerations associated with the development of model geometry, mesh generation, simulation, postprocessing, and validation of results, which is an integral practice in the appropriate progression of simulating and validating large production-type problems. The objectives set forth are accomplished by thorough evaluations of airflow developments through a fully-populated fuel bundle segment including scoping runs gauging resource requirements, grid independence, periodicity and symmetry applicability, validity of viscous model options, appropriateness of boundary conditions, and other pertinent setup characteristics establishing a sufficient level of accuracy and confidence in the final model implementation and acquired solution(s). Accordingly, the knowledge and understanding of the initial flow-field solutions gained from these computational trials are now incorporated into an extended model of the assembly flow domain encompassing an entire fully-populated fuel bundle run and spacer segment of single-span length as subsequently addressed.

5.2 SINGLE FULLY-POPULATED FUEL BUNDLE RUN AND SPACER

The ensuing research efforts are focused on the intent to provide accurate flow-field solutions to the substantially more-complex flow domain associated with an extended model segment of the fuel assembly, which includes the addition of a fully-populated spacer component adjacent to a fuel bundle run (both considered at full-length span). This computational flow domain is particularly significant towards gaining a more-complete understanding of the assembly airflow response since the internal geometry comprises a periodic flow segment of repeated component connections across

the fuel assembly structure that account for over one-half the total axial length. As a result, flow-field solutions of this modeled domain in conjunction with a comparative analysis of concluding hydraulic data from SNL afford potential answers to questions concerning initial geometric assumptions implemented in the spacer model representation, capability of the computational solver to resolve complex flow fields, adequate mesh construction with joint simulation setup, and current/future computational resource requirements necessary. Furthermore, the numerical trial results and postprocessed graphics for flow-field parameters are expected to provide SNL with a collection of hydraulic solution data that possesses meaningful benefit toward calibration efforts of the MELCOR SFP model. Pertinent details pertaining to the computational experiments performed and the overall conclusions drawn utilizing this domain model are presented in the following discussions.

5.2.1 Initial Considerations

The general construction of solid surface geometry representing the flow domain contained within this fully-populated bundle run and spacer segment model is sufficiently described in Chapter 4, which includes simplifying assumptions incorporated for the mixing devices located on spacer components. Again, aside from these spacer design simplifications, the dimensions are based on exact measurements specified in available manufacturer drawings provided by SNL for the GNF 9x9 BWR (GE 11 BWR/2-3) experimental fuel assembly and, where applicable, numerous other physical measurements for the assembly components with incomplete/absent drawings. These dimensions are not necessarily congruent with geometric values reported by Durbin and Lindgren (2005) in the hydraulic analysis details of the final experimental apparatus built

at SNL due to limitations previously discussed in Chapter 4; the existence of several slight variances in the measured dimensions is noted. The CAD solid model of the fully-populated bundle run and spacer flow domain, which spans the length of a fuel assembly segment between the midpoints of two successive fuel bundle runs and encompasses a single spacer component, is shown in Figure 5.49 below as an additional reference.

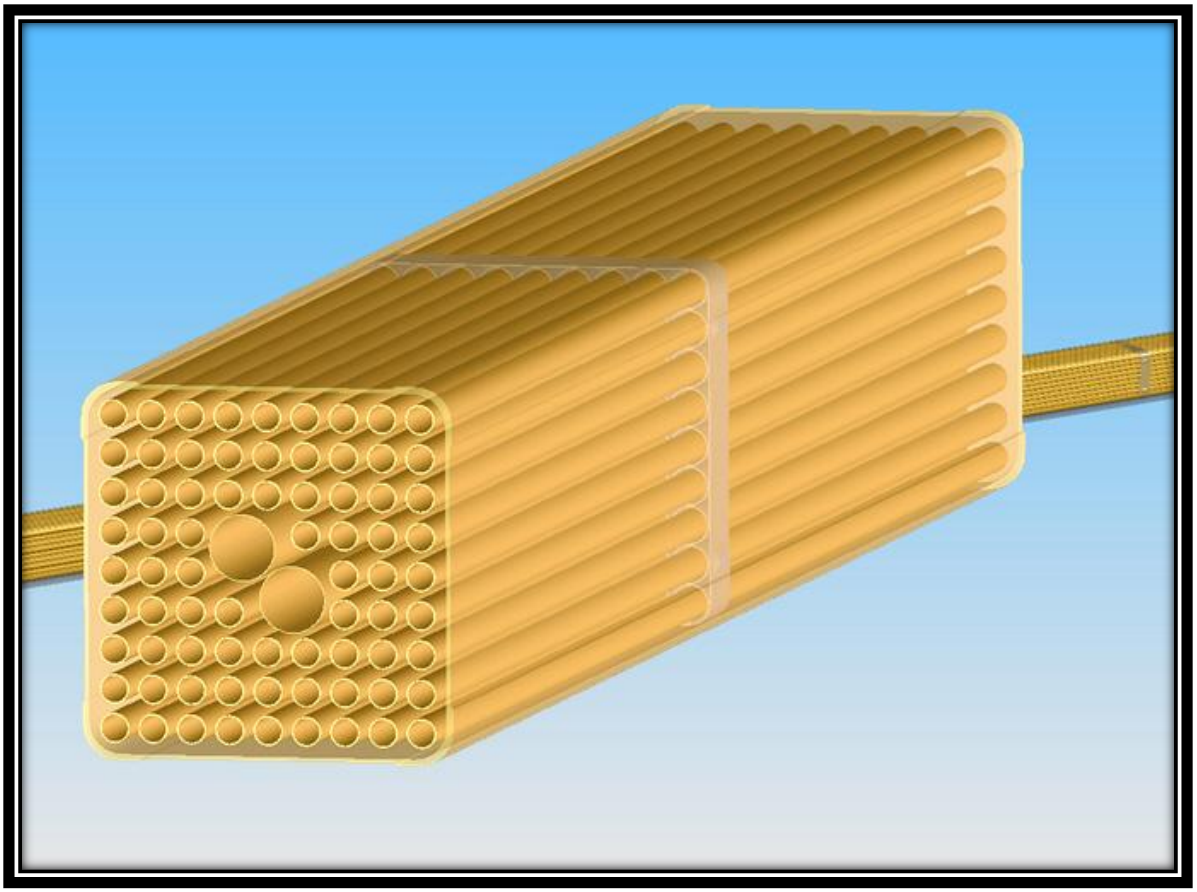


Figure 5.49: CAD solid model representation of fully-populated bundle run and spacer flow segment examined in computational trials; note that fuel and water rod elements are filled (or capped at ends) in actual flow domain model.

Fabrication of the domain grid follows the general considerations and principles outlined in Section 4.2.3 above for generating a desirable mesh characterized by minimal

numerical diffusion, truncation error, and stability/convergence difficulty while providing accurate resolution of the flow-field parameters and corresponding solution gradients. For the long narrow-ducted geometry and small hydraulic diameter associated with this flow domain model, the desired mesh attributes consist of a highly orthogonal element structure (as best aligned with anticipated axial flow), sufficient clustering of nodes/cells in areas of steep flow gradients, and high-quality mesh element construction. These attributes are addressed by features incorporated into the mesh development based on knowledge of flow-field solutions attained from the initial trials in Section 5.1, an objective to represent all interstitial flow spaces between wall surfaces with ten or more cell elements if practical (necessitating a minimum of five), and a thorough assessment of quality metric values.

The grid quality metrics of interest include equiangle skew (EAS), aspect ratio (AR), and size change (SC) for comprising volume cell elements. Specifically, the development of a high-quality 3D mesh is assured by achievement of guideline values for quality metrics of equiangle skew (*i.e.* $EAS_{max} < 0.5$ and $EAS_{avg} < 0.4$) and aspect ratio (*i.e.* $AR_{max} \leq 10$ for boundary or $AR_{max} \leq 5$ for core flow regions) that typify high-quality mesh elements, while maintaining reasonable size change values across element volumes (FLUENT Inc., 2006). However, one notable exception to the aspect ratio metric does exist for a limited number of volume cells near outer spacer lobes, which warrant aspect ratios greater than the guideline values in order to avoid unnecessarily large node/cell densities in these relatively benign flow areas due to the acute angle formed between sloping lobe surfaces and canister walls. These particular quality metrics are appropriately suited for assessing the alluded form of this final domain mesh, which is composed entirely of quadrilateral/hexahedral cell elements and generated using only the Map, Submap, and Cooper scheme types for volume meshing.

Significance of the aforementioned, as described in Sections 1.4 and 1.5, stems from difficulties limiting the current extent of research completed in this phase of modeling for the fuel bundle run and spacer flow segment. These difficulties (further evident in forthcoming deliberations), which are commonly encountered in research efforts entailing comparable large-scale simulations of complex flow domains, include extensive high performance computing requirements, substantial time allocations for geometry development and mesh generation, exorbitant uncertainty analyses and postprocessing for resulting data visualization, and large initial resource investments with slow solution turnarounds at the outset. The intricate fuel bundle and spacer component surfaces constitute a wide range of geometric and hydraulic length scales throughout the flow domain that are palpably consistent with the above conditions. Clearly, a reasonable understanding of the circumstances implies that several modeling tradeoffs concerning reduced focus on uncertainty assessments and model accuracy/verification analyses are requisite in order to fulfill a sensible scope of research and impart a practical array of meaningful results at the present stage of modeling efforts.

Therefore, consideration for the enormity of diverted resources from other research phases that are necessary in terms of additional time allocations and computing costs to facilitate even modest uncertainty assessments (or model validations) for the large-scale simulation trials ultimately precludes any resource intense analyses outside of the base computational experiments. In particular, the grid-independence study, multiple viscous model and solver setup analyses, and extensive postprocessing of high-end graphical results for each trial performed are unfeasible considerations for plausible completion within resource and scheduling constraints of the current research phase. The general results and nature of airflow responses from the fuel bundle run scoping studies as well as corresponding model validation assessments are assumed to reasonably reflect

the greater flow field of the fuel bundle with spacer segment domain and extend aid for acceptable development of this larger-scale simulation model. Thus, implementing knowledge of flow-field solutions gained from the preliminary experiments involving the bundle run model and adhering strictly to guideline values of relevant quality metrics for high-quality mesh construction are both extremely important aspects of this mesh model development and the effort to substantiate model validity.

Despite simplifications to the modeled flow domain geometry through the exclusion of mixing devices located on spacer components (detailed in Section 4.2), the formulation of a high-quality quadrilateral/hexahedral mesh with sought quality metric values and features described above is certainly not without its impediments. In order to ensure that adequate cell distributions are conservatively established throughout the computational model, the mesh generation process is initiated from within the spacer volume region, which represents interstitial spaces comprising the smallest flow areas across the fuel bundle run and spacer assembly segment and then proceeds outwards toward the inlet/outlet boundaries. Achieving a pervasive hexahedral embodiment of volume cells enveloping this region requires a substantial volume decomposition method that begins with differentiation of all singly-connected path segments traversing the spacer component into separate subvolumes from which the entire flow volume is composed. These subvolumes are further partitioned into two or more smaller constituent volumes allowing for a higher quality placement of hexahedral cells in the locality if deemed necessary.

Subsequently, upon generating a final grid for each interstitial volume entity within the spacer segment, meshed faces at both upstream and downstream cross-sectional interfaces between the fuel bundle and spacer flow volumes are utilized to extrude volume cells across the remaining fuel bundle halves of the model. The spacing

intervals incorporated in creating the volume mesh of both fuel bundle halves are varied along the axial length between spacer and model boundaries in an effort to reduce the total cell count. This is accomplished by maintaining appropriate intervals of reduced spacing in the proximity of spacer and inlet/outlet boundaries where higher resolutions are needed and/or flow gradients are expected, but slowly augmenting the intervals over increased distances from these boundaries.

Finally, due to detrimental memory limitations associated with the 32-bit platform upon which GAMBIT database files are built, the final mesh model is actually composed as a conglomerate of 30 individual mesh files representing the flow domain. Conformal grid interfaces are preserved across conjoining mesh file boundaries by creating each successive mesh from adjacent face cells at the end boundary of a preceding grid file construction. This permits fusing of appended mesh file volumes at individual interfaces such that every face cell in the opposing mesh file is accurately matched at the conjoining boundary.

Complete assembling of the approximate 260 million-cell mesh is performed by running a single serial FLUENT process on a high-performance visualization Sun system located at the Texas Advanced Computing Center (TACC) on The University of Texas at Austin campus that possesses 512 GB of accessible RAM from a central memory pool bank. The final appended mesh required approximately 482 GB of total memory to compile and setup an initial base model file from all grids using the FLUENT software, which is consequently suitable for implementation into the parallel-solver simulations. In summary, the pertinent geometric dimensions as well as grid characteristics of this final mesh model development are provided in the form of tabulated numerical values and descriptions including graphical depictions of the meshed flow domain in Table 5.8 and

Figures 5.50 through 5.52 below, respectively, which serve as further reference to the details imparted in above discussions.

Table 5.8: Reference geometry and mesh construction details of fully-populated bundle run and spacer final model.

GEOMETRY AND MESH CONSTRUCTION DETAILS OF FULLY-POPULATED FUEL BUNDLE RUN AND SPACER MODEL

Reference Model Geometry And Dimensions

[Component Cross-Section Data]

Component:	Spacer (Midplane)	Spacer-Lobe (Midplane)	Canister	Water-Rod	Fuel-Rod
No. Entities:	1	8	1	2	74
Area (m^2):	0.00242080	NA	NA	0.00048578	0.00009694
Perimeter (m):	6.80924734	NA	0.51550467	0.07813141	0.03490309
Diameter (m):	NA	NA	NA	0.02487000	0.01111000
Contact-Length (m):	NA	0.004200	NA	NA	NA
Pitch (m):	NA	NA	NA	NA	0.01437640
Bounded Area (m^2):	NA	NA	0.01783366	NA	NA

[General Flow Segment Data]

Flow Segment:	Spacer	Fuel Bundle Run	Entire Model
Length (m):	0.030480	0.481330	0.511810
Flow Area (m^2):	0.00726749 (Midplane)	0.00968829	NA
Wetted-Perimeter (m):	9.99664381 (Midplane)	3.25459647	NA
Hydraulic-Diameter (m):	0.00290797 (Midplane)	0.01190721	NA

Reference Model Mesh Characteristics

Total No. Volume Cells:	Face Elements:	Volume Elements:	Volume Mesh Scheme Types:	Max EAS Value:	No. Individual Mesh Files:	Final Case Mesh File [RAM] Required (GB):
≳ 260,000,000	Quadrilateral	Hexahedral	Map, Submap, Cooper	< 0.50	30	Approximately 482

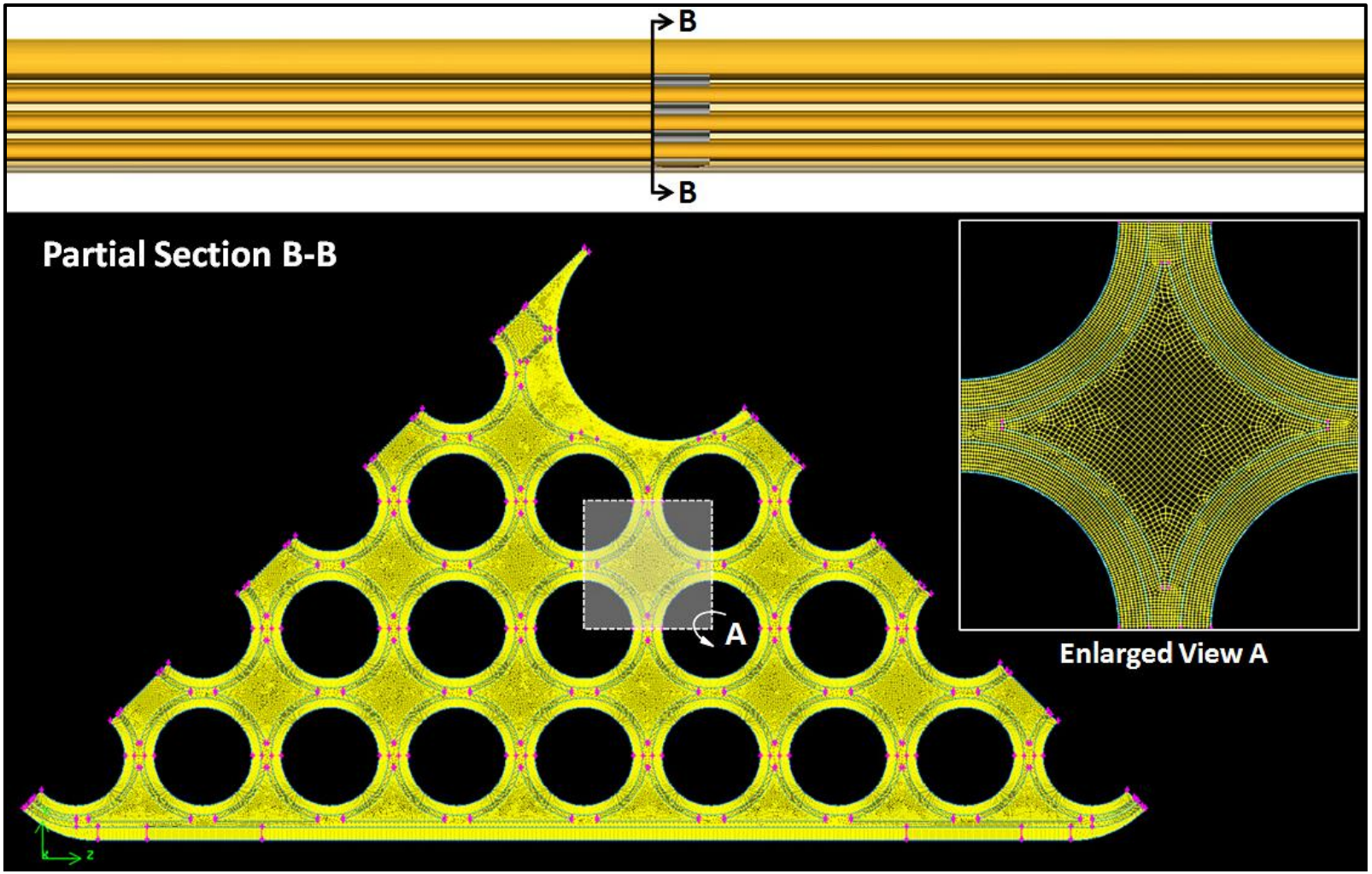


Figure 5.50: Depiction of mesh face cells at upstream fuel bundle and spacer interface cross-section.

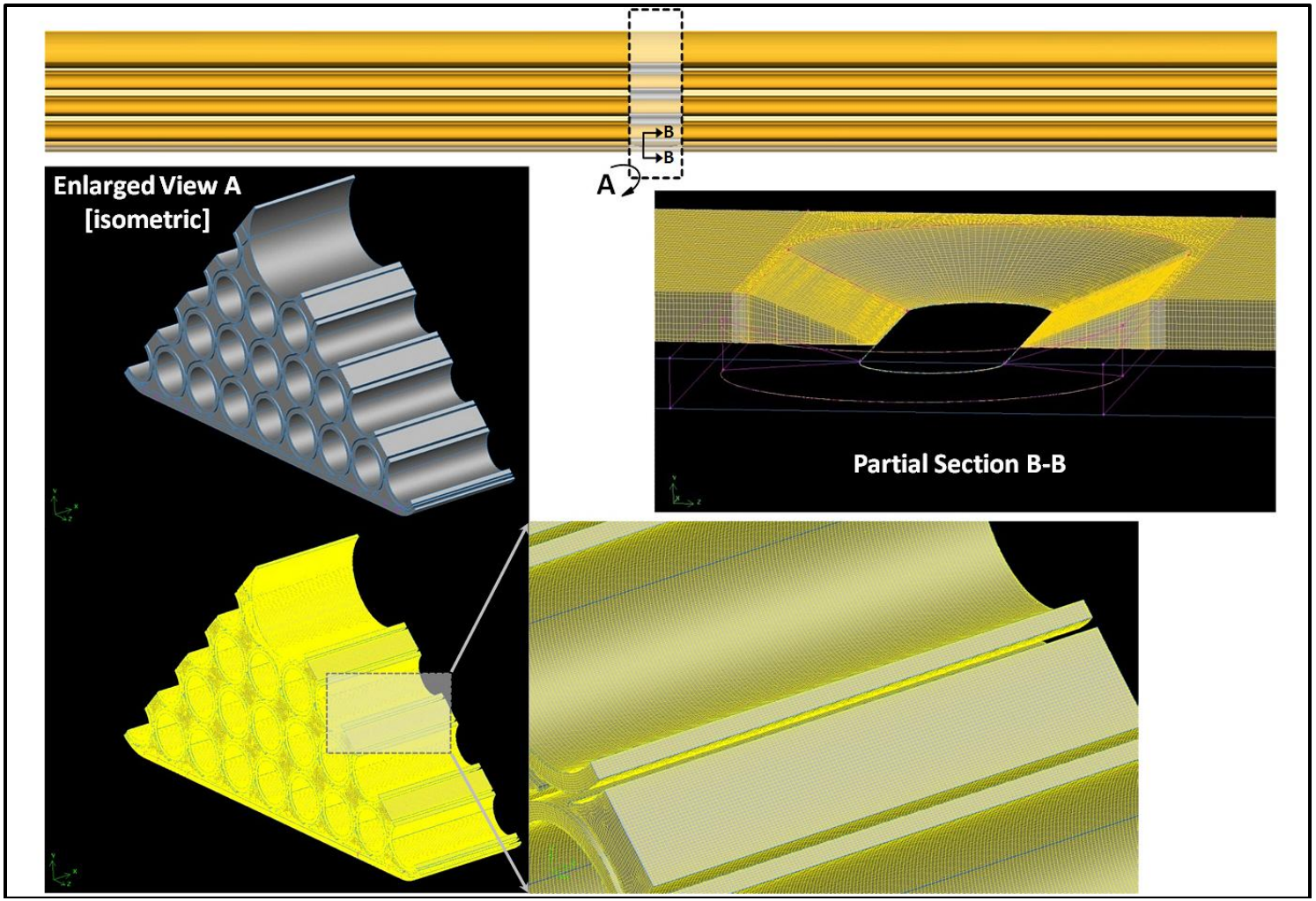


Figure 5.51: Depiction of mesh cell distributions across the spacer flow volume with emphasis on several enlarged regions.

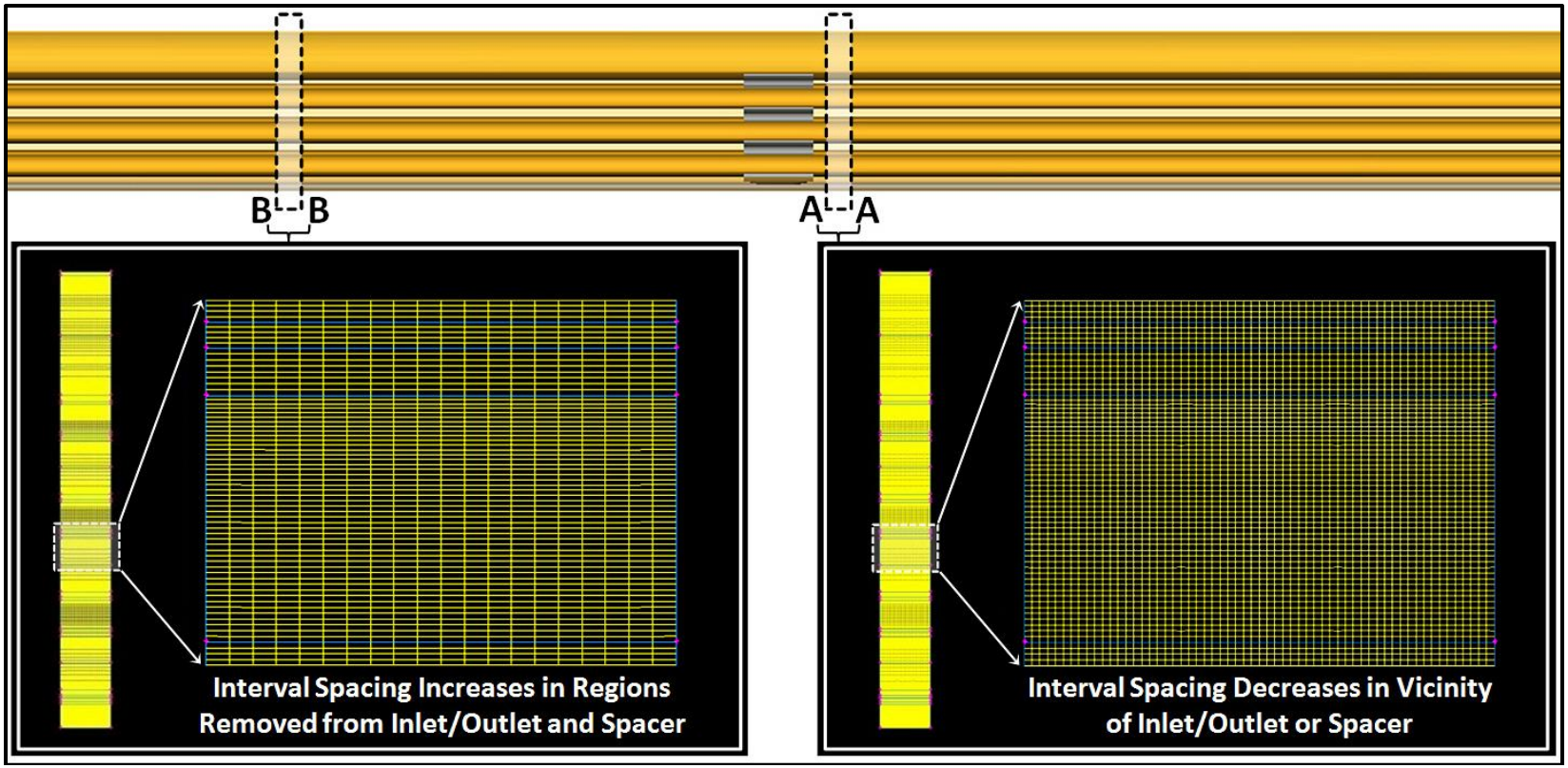


Figure 5.52: Depiction of interval spacing variations along axial length of model for mesh generation of bundle run segments.

Initial considerations for the computational trials performed below warrant additional explanation of two noteworthy specifics surrounding the viscous model option and inlet boundary condition selections employed in the model setup. The final comparison of results from Section 5.1.7 revealed evidence supporting both the standard k-ω transitional and laminar model types as sufficient for capturing related airflow responses through a fuel bundle segment. However, the laminar viscous model option is incorporated in the subsequent numerical experiments due to the reduced load on computational resources required for each simulation.

Although a thorough comparison of results justifying the adequacy of utilizing the laminar viscous model over the former option is absent from concluding solutions presented below, these investigations did include two preliminary trial runs at the highest flow rate examined (*i.e.* 600 slpm) to assess viscous influences (if any) of the spacer addition to this domain model not previously anticipated. A contrast of pressure losses across the entire flow model predicted by both viscous model types at the highest airflow confirmed that reasonably equivalent solutions are attained using either solver setup with less than a 3% margin of difference between the solution values. Hence, ample substantiation of the assumption that an applied laminar viscous option affords relatively indifferent results for this particular mesh model in comparison with the more resource intensive standard k-ω transitional selection is addressed.

Concerning the inlet boundary condition exercised in these simulations, the final parameter flow-field values reported in Section 5.1.7 also illustrate a significant probability for development of a repeating, highly periodic nature of flow (exhibited by a consistent pressure drop and velocity field) across a majority of the multiple bundle run and spacer segments comprising the lower, fully-populated fuel assembly region. This implies that an assumption of flow periodicity between the domain inlet and outlet

encompassing a fully-populated bundle run and spacer of single-span length imparts meaningful results for a comparative analysis with SNL experiments by simply enforcing periodic boundary conditions in the model setup. Unfortunately, due to the aforementioned procedure required for assembling the final mesh construction, incorporating an accurate set of matching face meshes for paired inlet/outlet boundary planes proved extremely difficult and unworkable under the stated circumstances, which is a necessary constraint for establishing congruent periodic boundaries in the FLUENT CFD software.

Therefore, in order to obtain solutions characterizing a nearly periodic flow-field development traversing the modeled domain segment, a viable alternative method is requisite for enabling the periodicity of flow by exploiting different boundary condition implements available. Ultimately, this is accomplished through looping of the outlet velocity profile from a preceding run to the velocity inlet of an ensuing run as a profile boundary condition, beginning with a uniform velocity inlet for the initial loop of each flow rate evaluated. This process of looping the outlet velocity profile to the velocity inlet of a subsequent run is repeated until the percent difference in predicted pressure loss across the flow domain between successive trial loops is less than 1% for all airflow rates. At this point, it is assumed that reasonable estimates of the periodic flow fields associated with the entire range of flow rates investigated are established within sufficient limits of numerical accuracy.

5.2.2 Looped Inlet Velocity Profile Runs

The five airflow rates selected for examination cover a broader range of values compared with prior studies involving the fuel bundle run in Section 5.1, which span

uniform flow rate intervals from 200 to 600 slpm. Moreover, the general solver and model options within FLUENT for parallel processing of each computational trial follow a basic skeletal outline as previously discussed in Section 5.1 for the concluding bundle run simulations as well. These and other pertinent details describing the essential model setups utilized in corresponding numerical experiments performed are briefly summarized below.

Key Setup Features of Computational Model:

- The 3D, double precision, steady, pressure-based solver formulation is implemented for all simulations with the laminar viscous model and gravitational acceleration options selected.
- Solution methods include the SIMPLE pressure-velocity coupling scheme as well as Green-Gauss node-based gradient, second order pressure, and second-order upwind momentum options for spatial discretization.
- Partitioning of the computational grid for launched parallel-computing processes within FLUENT is achieved by applying the METIS partition method.
- Defaults are maintained for under-relaxation factors, equations solved, and all fluid cell zone as well as wall boundary conditions.
- Two symmetry plane boundaries are employed in the model as shown above.
- Inlet boundary conditions consist of a uniform velocity specification for the initial loop and a velocity profile for all subsequent loops at each flow rate investigated as previously discussed. A pressure outlet at ambient conditions is always exercised in these simulations.

- Convergence criterion for scaled residuals is $1x10^{-5}$ for all equations. Supplemental convergence monitors include the mass imbalance and AWA static pressure differential across the inlet/outlet model boundaries.
- The fluid airflow is incompressible with constant properties, but the specified values are slightly different as compared to averages reported in the hydraulic analysis summary by Durbin and Lindgren (2005) for the SNL experiments. For consistency with the heated, buoyancy-induced flow simulations addressed later in Section 5.3, an approximate average for the ambient atmospheric pressure and temperature values over a few of the disclosed experimental data set collections provided by SNL is used with the incompressible ideal gas law to calculate the air density value implemented in these trials.
- Additionally, dynamic viscosity is based on the same tabulation of numerical values for a range of air temperatures from Incropera and DeWitt (2002), which are also employed in the buoyancy-induced computational runs of Section 5.3 below. These subtle air property variations are assumed to incur a minimal impact of detrimental influence to the overall final comparison of derived loss coefficient results with the hydraulic analysis values determined from SNL experimental measurements.
- Five flow rates are examined in the ensuing simulation trials that include airflows of 200, 300, 400, 500, and 600 slpm.

In order to facilitate and efficiently acquire resolved flow-field solutions on this large-scale grid model, an extensive high-performance, parallel-computing resource is required. Ranger, a Sun Constellation Linux Cluster at the TACC possessing 62,976 processing cores and a peak performance of 579.4 TFLOPS, affords this necessary

computational capability and is one of the largest computing resources in the world (Texas Advanced Computing Center, 2010). Furthermore, every parallel-process iterative calculation submitted to this system is performed in a batch-mode (or background) processing environment, which is controlled by a SGE job script file and a second run script file containing the FLUENT text commands executed during the scheduled run.

A desired balance between available software licenses, computational efficiency, and memory requirements per compute node is represented by the following requested resource configuration specified for all simulation trials: (1) total allocation of 25 compute nodes encompassing 400 processing cores; (2) on master compute node only the FLUENT host process with a single correlating MPI task is launched; and (3) on each of the 24 remaining children compute nodes 8 MPI tasks are launched. This configuration allots adequate memory for the FLUENT processes launched on every compute node and results in a total of 193 FLUENT MPI tasks with 800 GB of aggregate memory. The number of iterations required for solution convergence varied from approximately 2000 to 3500 in the individual velocity loop trials conducted.

On a final note, the FLUENT CFD software offers several different surface integral options for postprocessing the flow-field pressures including both area- and mass-weighted averages (*i.e.* AWA and MWA) of static as well as total pressure values. Unlike the previous preliminary trials (as subsequently detailed), the type of pressure quantities selected for postprocessing and comparative analysis with SNL experiments concerning this flow domain model is of utmost importance, especially with respect to the curve fitting of solution data as outlined for the hydraulic analysis procedure. Additional insight into justification for the opted type of pressure result in these

numerical simulation comparisons is provided via considerations of energy conservation associated with steady flow across a general piping system.

In a simple incompressible flow through a control volume pipe, which is similarly analogous to the airflow traversing a related fuel assembly segment, one potential form of the energy equation in terms of energy per unit mass that directly correlates with the definition of total head loss (h_{lT}) between two cross-sectional plane locations is given by (Fox and McDonald, 1992):

$$\left(\frac{p_1}{\rho} + \alpha_1 \frac{\bar{V}_1^2}{2} + g z_1\right) - \left(\frac{p_2}{\rho} + \alpha_2 \frac{\bar{V}_2^2}{2} + g z_2\right) = h_{lT} \tag{5.30}$$

or, in terms of energy per unit volume (*i.e.* pressure), Equation (5.30) is expressed as

$$\left(p_1 + \alpha_1 \frac{\rho \bar{V}_1^2}{2} + \rho g z_1\right) - \left(p_2 + \alpha_2 \frac{\rho \bar{V}_2^2}{2} + \rho g z_2\right) = \Delta P_{total} \tag{5.31}$$

where, the kinetic energy coefficient (α) is defined according to

$$\alpha = \frac{\int_A \rho V^3 \, dA}{\dot{m} \bar{V}^2} \tag{5.32}$$

and, the local as well as average cross-section velocities are denoted by (V) and ($\bar{V}$), respectively. The total pressure loss (ΔP_{total}) stated in Equation (5.31) represents the sum of major and minor pressure losses between two piping system points (not necessarily a pressure differential for two measurement quantities of total pressure type).

This parameter value is formulated in the form of Equation (3.9) above for curve fitting the SNL experimental data as part of the hydraulic analysis process explained in

the final report by Durbin and Lindgren (2005). However, it is crucial to emphasize that the terms representing hydrostatic head due to elevation difference in Equation (5.31) do not require explicit consideration in corresponding calculations involving this expression since the SNL experimental measurements and default pressures reported by the FLUENT CFD software already exclude hydrostatic head contributions from all differential pressure computations. Again, this exclusion is requisite for designations of total head loss as such measurements only quantify the irreversible loss in mechanical energy (*i.e.* viscous and inertial loss effects) across control volume boundaries of the internal flow system. Thus, aside from hydrostatic head terms, the question remains as to which form of surface integral and type of pressure are best symbolized by the two groups of remaining bracketed terms (each representing the mechanical energy per unit volume at a cross-section) on the left-hand side of Equation (5.31).

Although the differential pressure is experimentally measured by SNL between two pressure ports using a set of plumbed pressure transducers (shown in Figure 3.1) that essentially characterize a difference of static pressure values, there is a concern with basing the flow losses across any assembly component of near single-span length on such a static pressure measurement. The issue arises predominantly as a result of improper consideration for the kinetic energy coefficient at both pressure port locations spanning a single assembly component, which is highly dependent upon the velocity profile at each point. For example, the static pressure differential across ports (11-13) for a fuel bundle run or ports (13-14) for a spacer (as experimentally measured by SNL), fails to account for the obvious variance in the kinetic energy coefficient values (clearly indicated by dissimilar velocity profile developments) between the ports.

In order to address this kinetic energy coefficient variance, these terms in Equation (5.31) necessitate the inclusion of a sufficient quantification for dynamic

pressure difference in addition to the static pressure, which is only realized through measurements of total pressure. Furthermore, the definition of (α) in Equation (5.32) suggests that a mass-weighted average is most appropriate for the surface integral values of total pressure, especially at moderate Reynolds numbers where the anticipated change of kinetic energy is reasonably large. Therefore, in summary, pressure losses represented by the left-hand side of Equation (5.31) above for two pressure ports traversing any single assembly component are adequately captured by only a differential pressure loss formulated from MWA total pressures taken at both port locations. The detrimental effect on solutions fit to the form of Equation (3.9), but based on static pressure loss data for the embodiment of terms in Equation (5.31) corresponding to a single assembly component (as implemented in the hydraulic analysis by SNL), is subsequently revisited in Section 5.2.3 below.

On the other hand, pressure losses associated with ports spanning multiple component segments of the fuel assembly denoting both a periodic geometry and nature of flow-field development are not susceptible to the above restrictions since (α) is equivalent at the two ports. In other words, for expectations of near congruent velocity profile developments at two pressure measurement points, the assumption of reasonably equivalent (α) values is permissible and AWA static pressures are appropriate for determining the differential pressure across such flow segments. Since the final comparison of results in Section 5.1.7 focused primarily upon a contrast of numerically tabulated solutions attributed to periodic flow with the hydraulic analysis values from SNL (shown in Tables 5.6 and 5.7), the above reasoning clarifies the use of differential static pressures as suitable for that particular comparison study.

Similarly, either the MWA total or AWA static pressures are sufficient for characterizing the pressure losses across the current flow domain model encompassing a

fully-populated bundle run and spacer of single-span length; but only for the entire model, otherwise MWA total pressures are required in designated shorter spans that involve one or more interior cross-section planes. This is because of the objective to acquire final flow-field solutions correlated with a periodic flow assumption for each airflow rate examined and thus (α) is theoretically identical at the inlet/outlet boundary planes under such conditions. However, to remain consistent in the presentation of all curve-fit solutions for the entire domain model as well as comprising component segments including those in forthcoming sections of this chapter and Chapter 6, results are foremost cast in terms of MWA total pressure values regardless of the domain extent evaluated.

Again, a process of looping the outlet velocity profile to the velocity inlet of a subsequent run is repeated until the percent difference in predicted pressure loss across the modeled domain is less than 1% between successive trial loops at all flow rates. This applies to both the MWA total and AWA static pressure loss quantities since flow periodicity over the entire model length implies a near equivalence and negligible deviation (in successive loops) between these two values as previously discussed. As a result, the differential pressures based on both MWA total and AWA static pressures are monitored for the 1% difference between successive trial loops to provide greater certainty of a near periodic flow establishment (within reasonable limits) for each airflow rate specified. The important characteristic references detailing the simulation model setup along with numerically tabulated results for the pressure losses of each successive velocity loop trial and airflow rate examined are included in Table 5.9 below, which is followed by a graphical plot in Figure 5.53 of the pressure losses for each velocity loop and flow rate evaluated.

Table 5.9: Reference values and pressure loss results over entire model length for all velocity loop trials.

INITIAL PRESSURE LOSS RESULTS AND SETUP CONDITIONS FOR FUEL BUNDLE RUN AND SPACER (FULLY-POPULATED) MODEL LOOP RUNS

Reference Model Values				Reference Component Average Velocity Values		
Air Density (kg/m^3)	0.98141470	Loop 1 - Inlet Velocity Profile	Uniform	Flow Rate (slpm)	Bundle Velocity (m/s)	Spacer Velocity (m/s)
Air Viscosity (kg/s-m) $_{dynamic}$	1.820975E-05	Other Loops - Inlet Velocity Profile	Prior Loop Exit Profile	200	0.45385897	0.60503955
Operating Pressure (Pa)	82833.7622	Total Model Length (m) $_{domain}$	0.511810	300	0.68078845	0.90755932
Gravity (m/s^2)	-9.81	Fuel Bundle Flow Area (m^2) $_{x\text{-}section}$	0.00968829	400	0.90771794	1.21007910
Temperature (K)	294.4034	Fuel Bundle Hydraulic Diameter (m) $_{x\text{-}section}$	0.01190703	500	1.13464742	1.51259887
Viscous Model	Laminar	Spacer Flow Area (m^2) $_{midplane\ x\text{-}section}$	0.00726749	600	1.36157691	1.81511864
Inlet/Outlet Boundary Types	V_{inlet} / P_{outlet}	Spacer Hydraulic Diameter (m) $_{midplane\ x\text{-}section}$	0.00290797			
Pressure Outlet Gauge Pressure (Pa) $_{outlet}$	0	Total Volume Cell Count $_{domain}$	≈ 260,000,000			

Pressure Loss Results And Loop Comparisons - Entire Model Length

[Pressure Losses Based On Area-Weighted-Average (AWA) Static Pressure Values]					[(AWA) Static Pressure Loss Differences Between Loops]			
Flow Rate (slpm):	P-Loss Loop 1 (Pa):	P-Loss Loop 2 (Pa):	P-Loss Loop 3 (Pa):	P-Loss Loop 4 (Pa):	Flow Rate (slpm):	Loops 1 - 2 (%):	Loops 2 - 3 (%):	Loops 3 - 4 (%):
200	2.557228	2.320029	2.312341	2.311066	200	9.2757	0.3314	0.0551
300	4.168477	3.642281	3.610290	3.605859	300	12.6232	0.8783	0.1227
400	5.956547	5.044814	4.973529	4.960547	400	15.3064	1.4130	0.2610
500	7.905115	6.518090	6.393230	6.364418	500	17.5459	1.9156	0.4507
600	9.998965	8.064576	7.857808	7.805170	600	19.3459	2.5639	0.6699

[Pressure Losses Based On Mass-Weighted-Average (MWA) Total Pressure Values]					[(MWA) Total Pressure Loss Differences Between Loops]			
Flow Rate (slpm):	P-Loss Loop 1 (Pa):	P-Loss Loop 2 (Pa):	P-Loss Loop 3 (Pa):	P-Loss Loop 4 (Pa):	Flow Rate (slpm):	Loops 1 - 2 (%):	Loops 2 - 3 (%):	Loops 3 - 4 (%):
200	2.392013	2.310112	2.310766	2.310558	200	3.4239	0.0283	0.0090
300	3.774694	3.598803	3.602200	3.604075	300	4.6597	0.0944	0.0521
400	5.236419	4.944367	4.949711	4.954734	400	5.5773	0.1081	0.1015
500	6.763740	6.337693	6.340765	6.348556	500	6.2990	0.0485	0.1229
600	8.351377	7.771323	7.761650	7.772161	600	6.9456	0.1245	0.1354

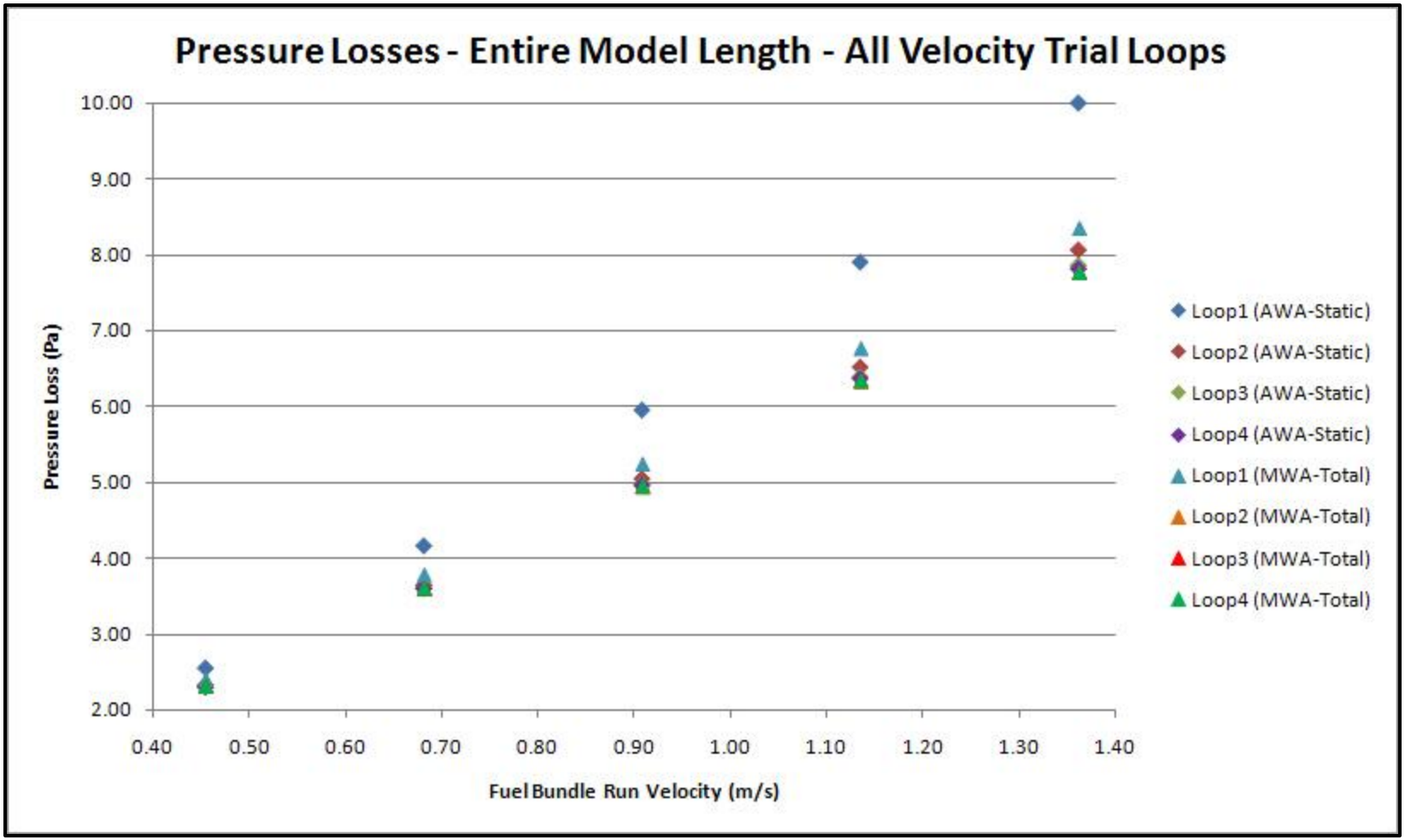


Figure 5.53: Graphical plot of pressure loss results over entire model length for all velocity loop trials.

Based on numerical pressure loss results shown in Table 5.9, four trials corresponding to three different loops of the velocity outlet profile from a previous loop as the new velocity profile specification for the inlet boundary condition of an ensuing loop are necessary to achieve the 1% difference requirement in predicted pressure losses between successive loops at each flow rate. Again, this 1% difference requirement applies to the predicted pressure differentials formulated from both MWA total and AWA static pressure values across the model inlet/outlet boundaries. As expected, the differences in predicted pressure losses between MWA total and AWA static pressure values decrease for all flow rates with each successive velocity loop. This stems from an increasingly closer approximation of periodic flow conditions with the pressures measured across model inlet/outlet boundary locations of anticipated near-steady flow development, which is illustrated in Figure 5.53 above.

The following postprocessed graphical results are predominantly based on selected cross-sectional planes and line profiles at several locations of particular interest within the modeled domain; these are defined in Figure 5.54 below. As previously discussed, the substantial cost in computational resources associated with the postprocessing of graphical results for this large-scale mesh model precludes an extensive presentation of significant solution parameters for all simulations conducted. Therefore, the postprocessing of graphics for numerical trials involving the fully-populated bundle run and spacer model is focused only on those parameter results attributed to the final velocity profile loop (*i.e.* loop 4) at each of the five airflow rates examined. These postprocessed solution graphics are depicted in Figures 5.55 through 5.80 below.

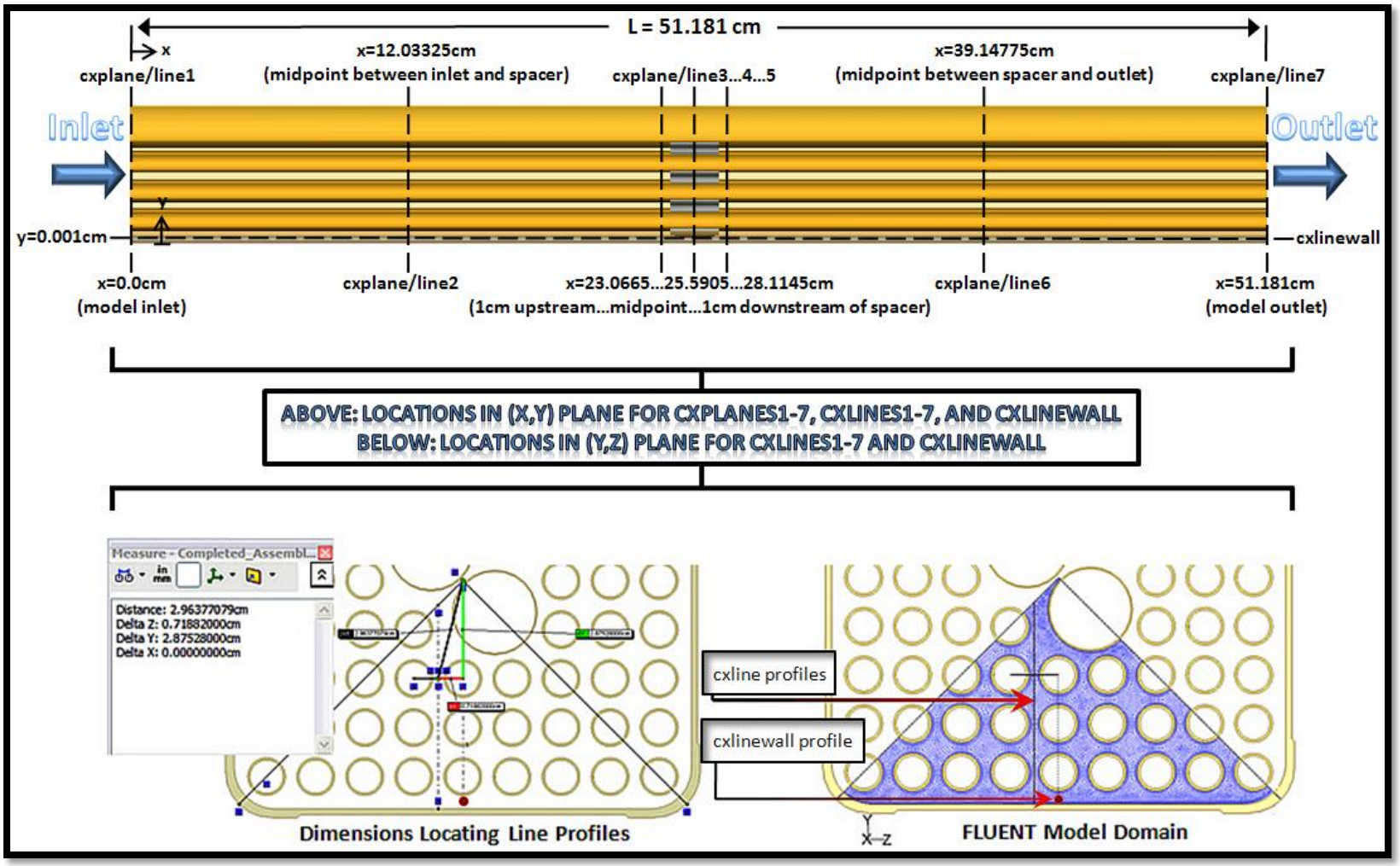


Figure 5.54: Defined locations of cross-sectional planes and line profiles referenced throughout postprocessed graphic results.

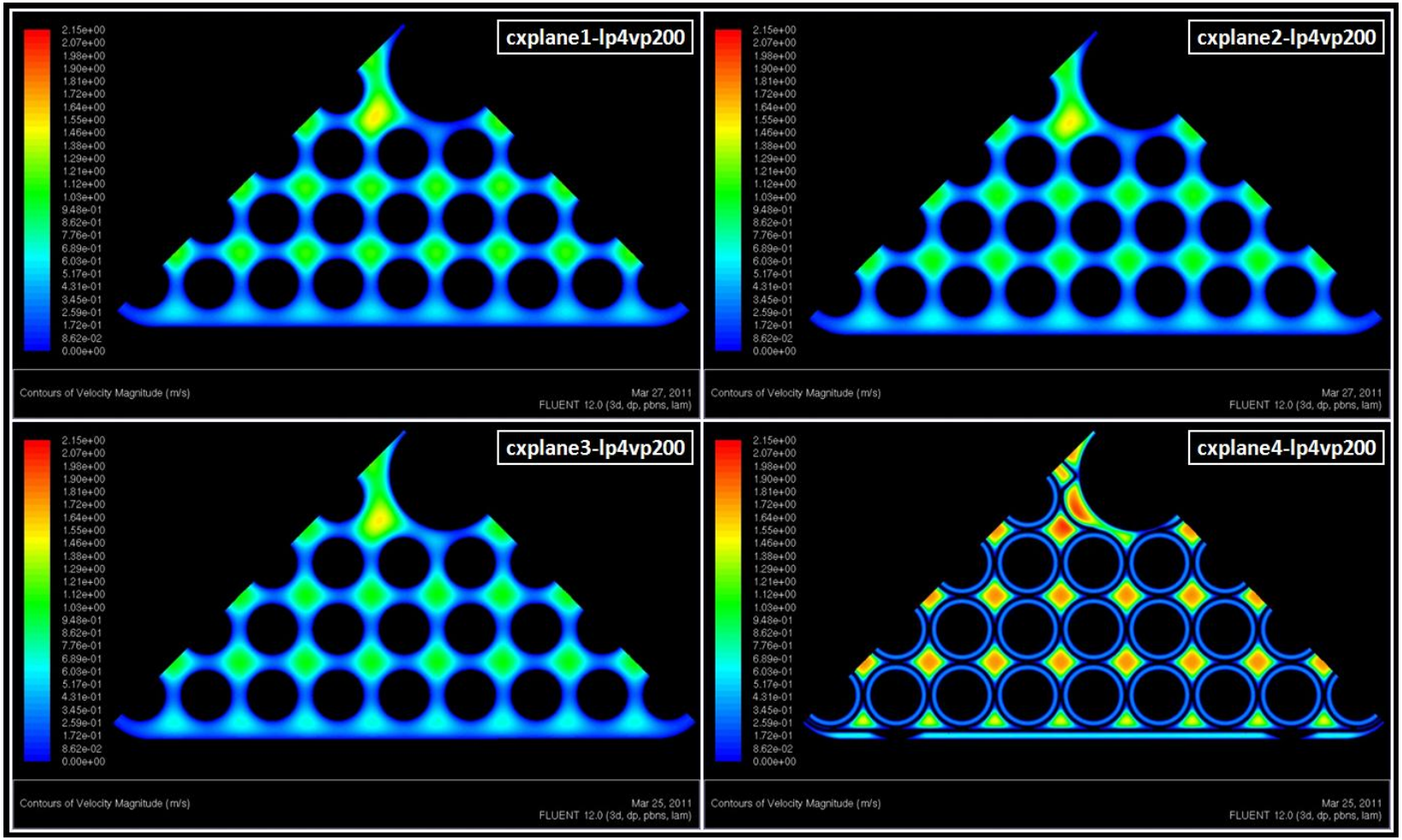


Figure 5.55: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 200 slpm (1 of 2); see image labels for details (*i.e.* 'cxplane1-lp4vp200' is cross-sectional plane 1 - loop 4 velocity profile at 200 slpm).

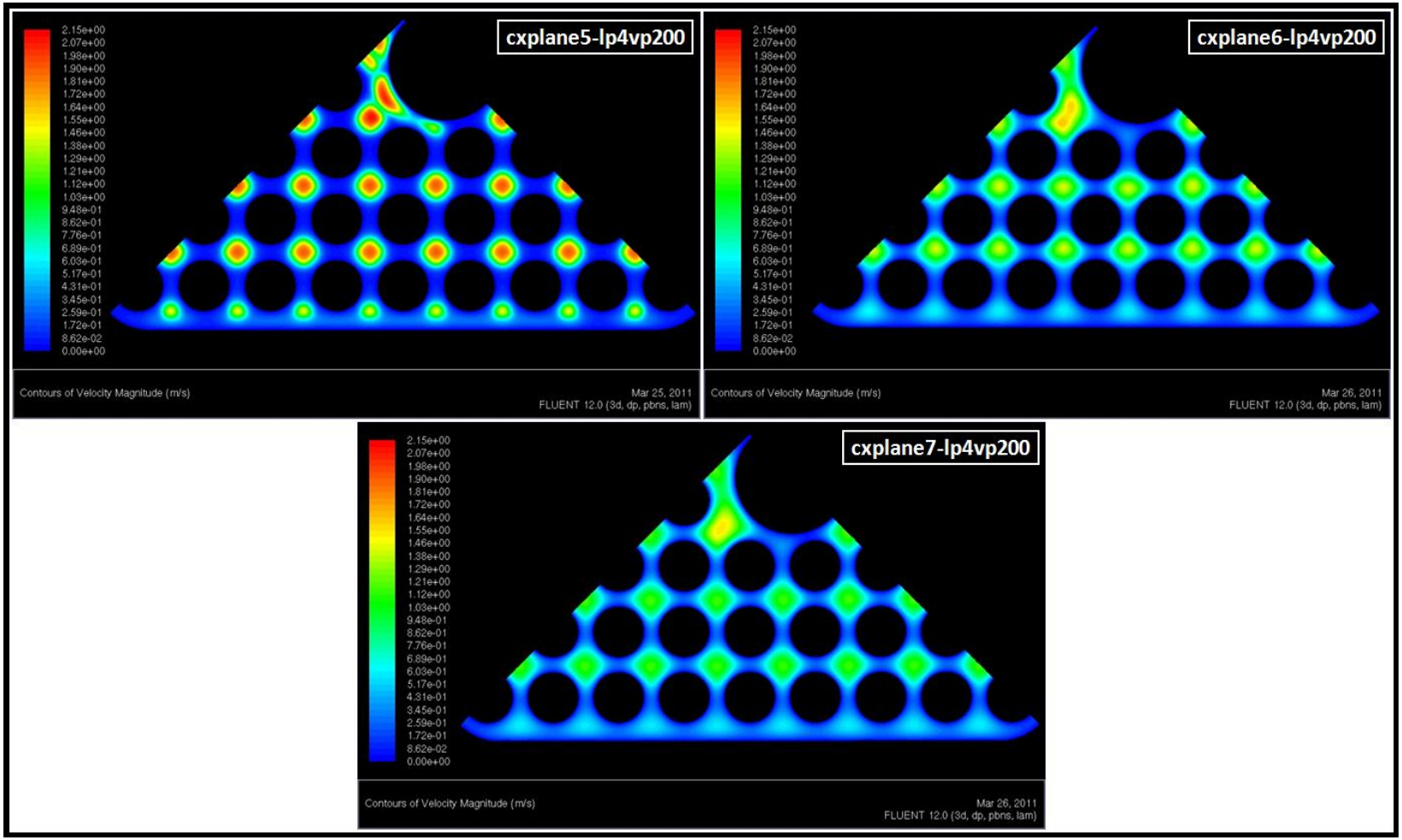


Figure 5.56: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 200 slpm (2 of 2).

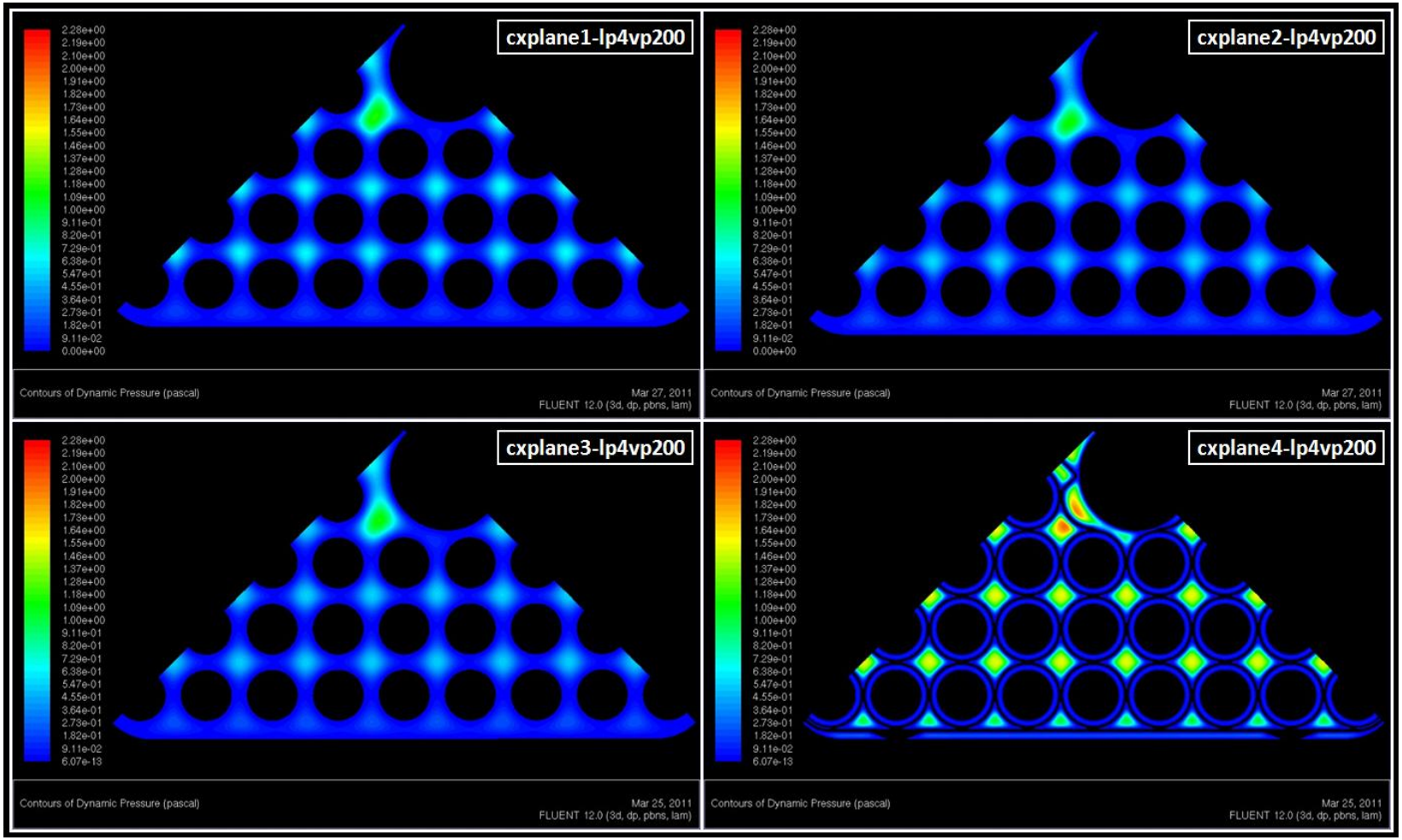


Figure 5.57: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 200 slpm (1 of 2).

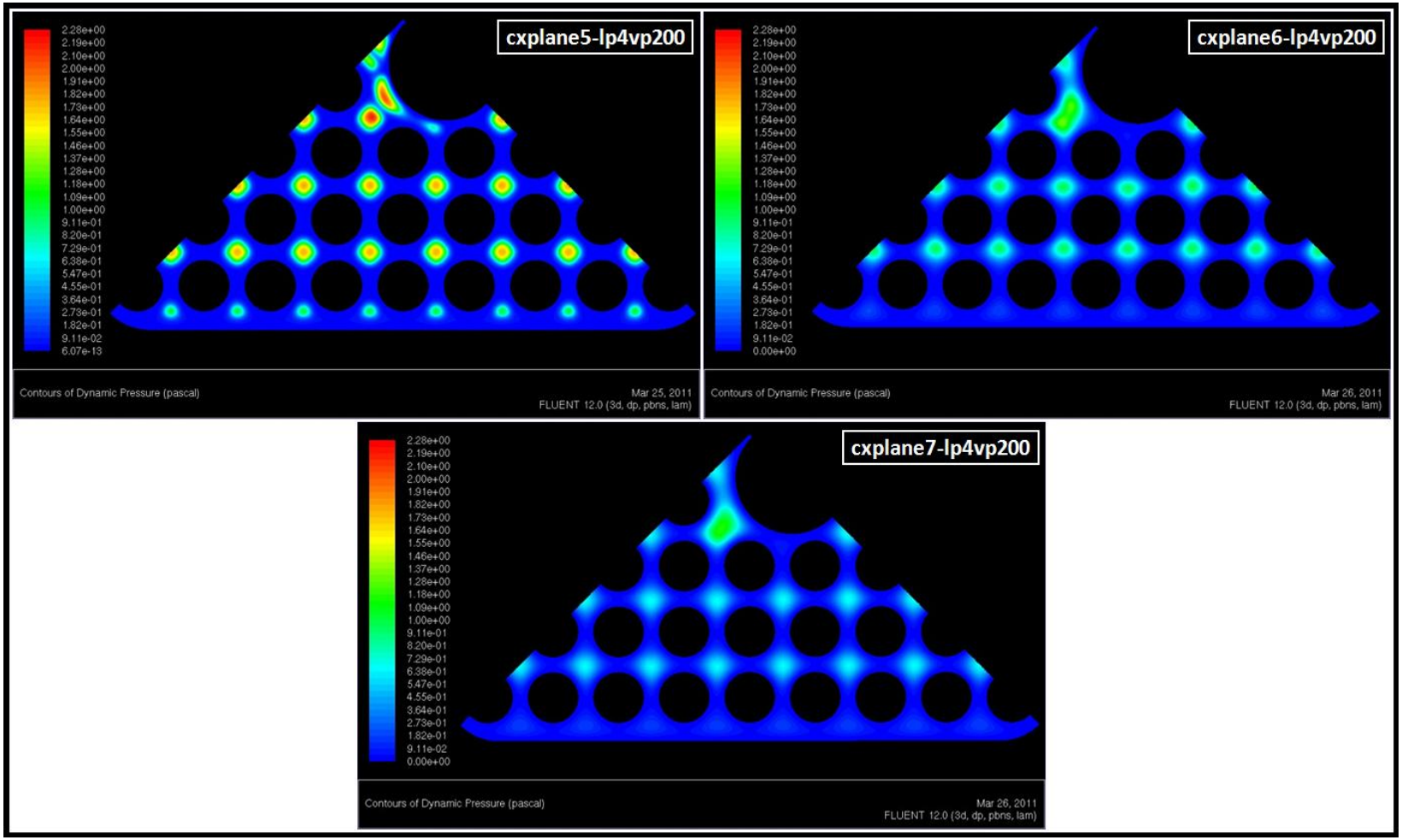


Figure 5.58: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 200 slpm (2 of 2).

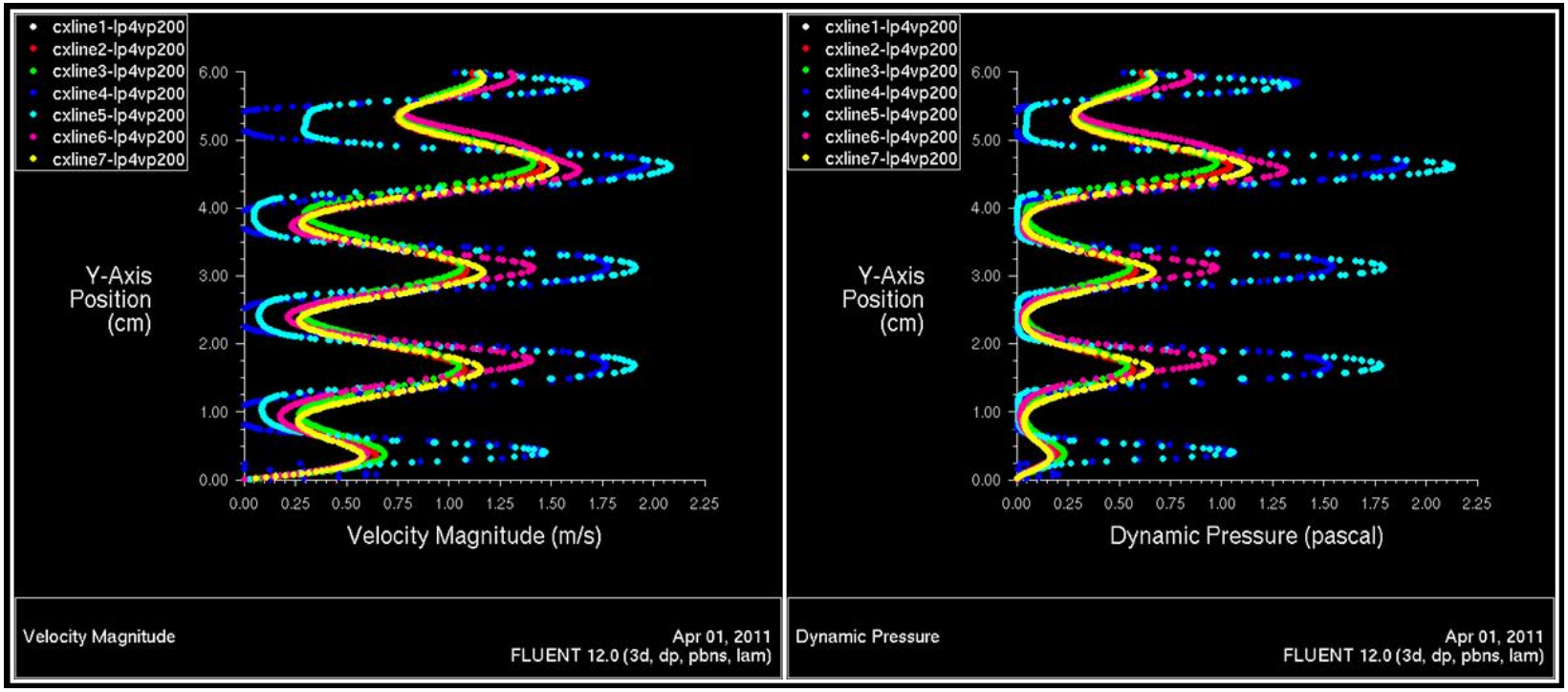


Figure 5.59: Velocity magnitude and dynamic pressure values for specified line surface profiles from loop 4 velocity trial at 200 slpm. Again, explicit details of data set entries are provided by the corresponding legend labels (*i.e.* 'cxline1-lp4vp200' is for line surface profile 1 - loop 4 velocity profile at 200 slpm).

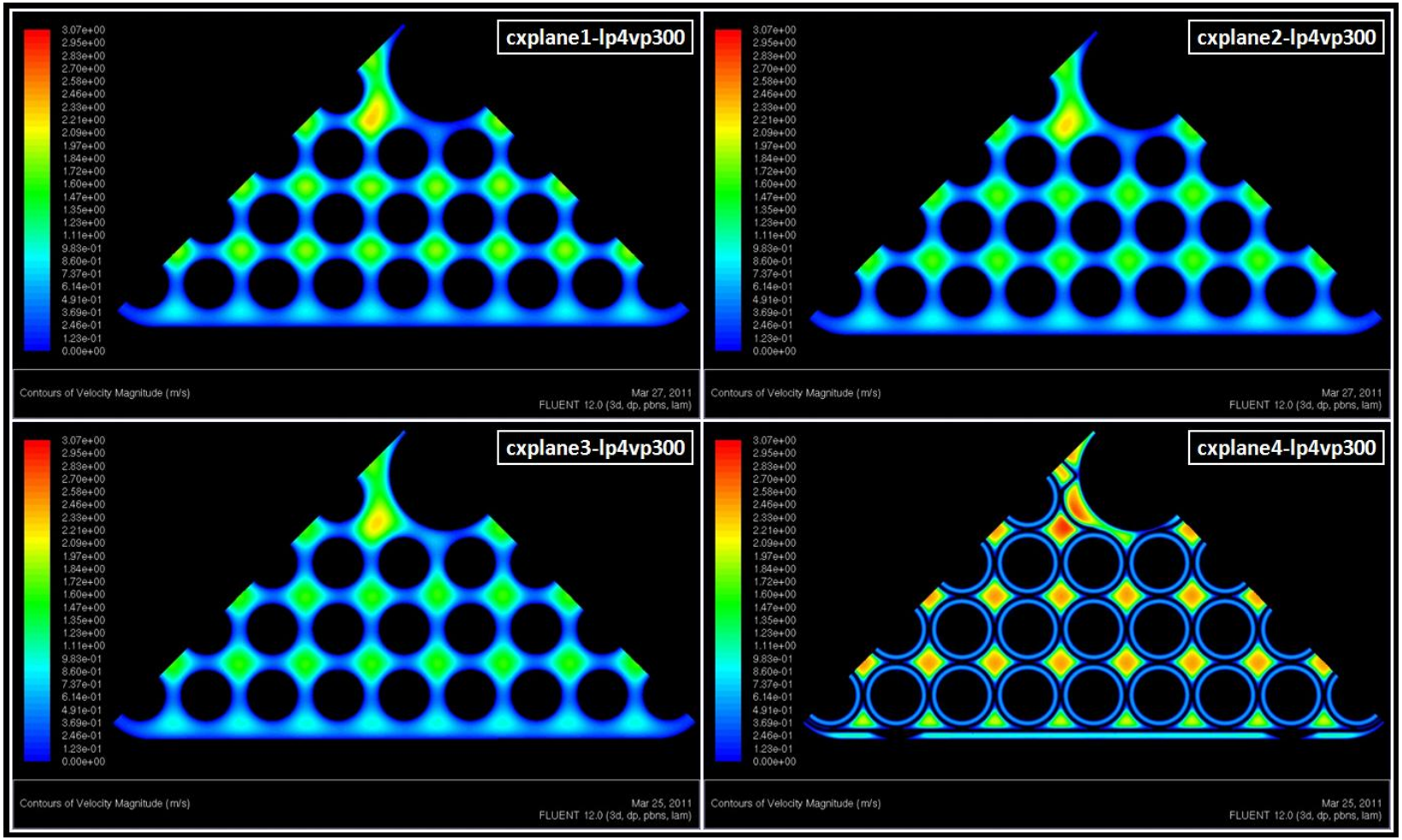


Figure 5.60: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 300 slpm (1 of 2).

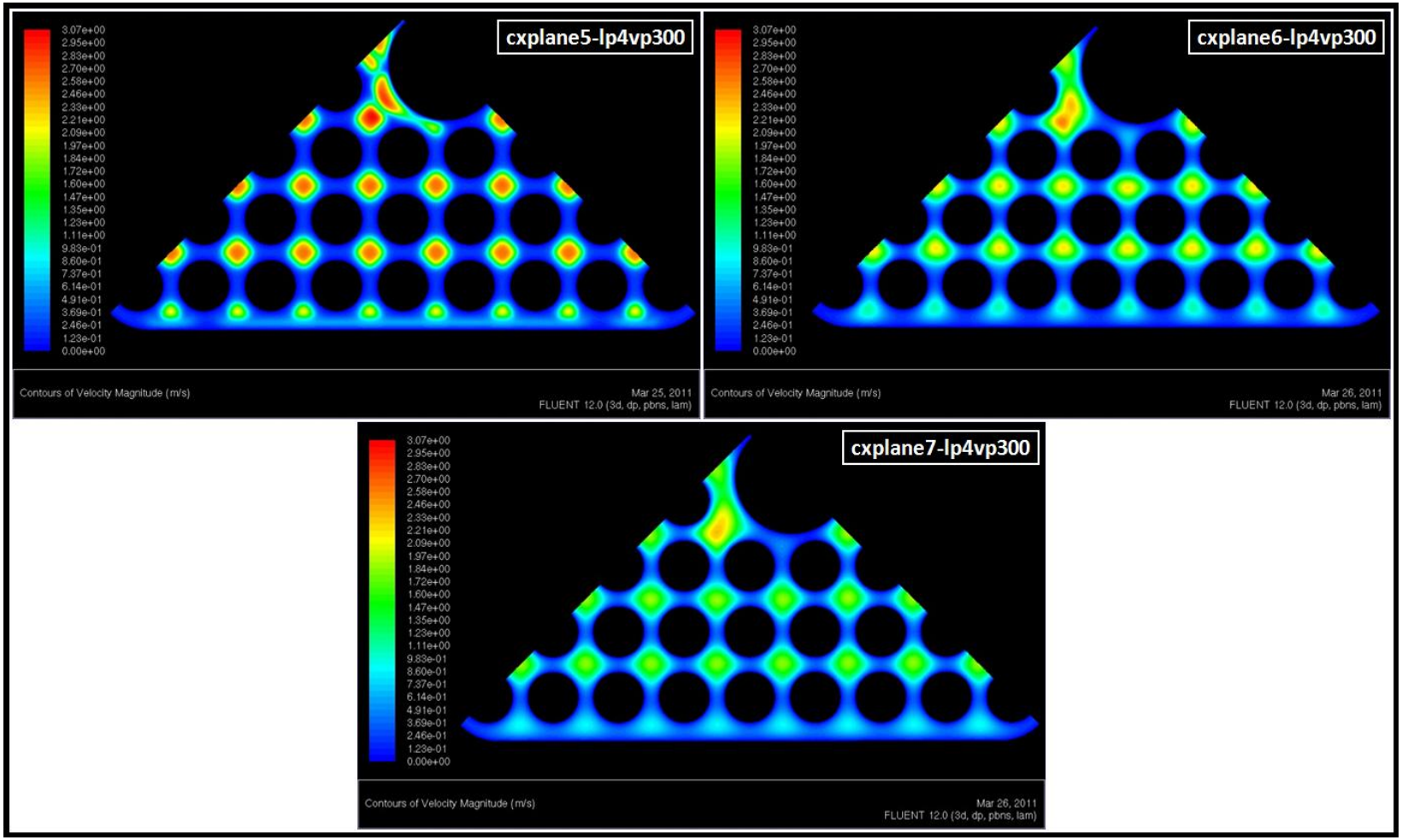


Figure 5.61: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 300 slpm (2 of 2).

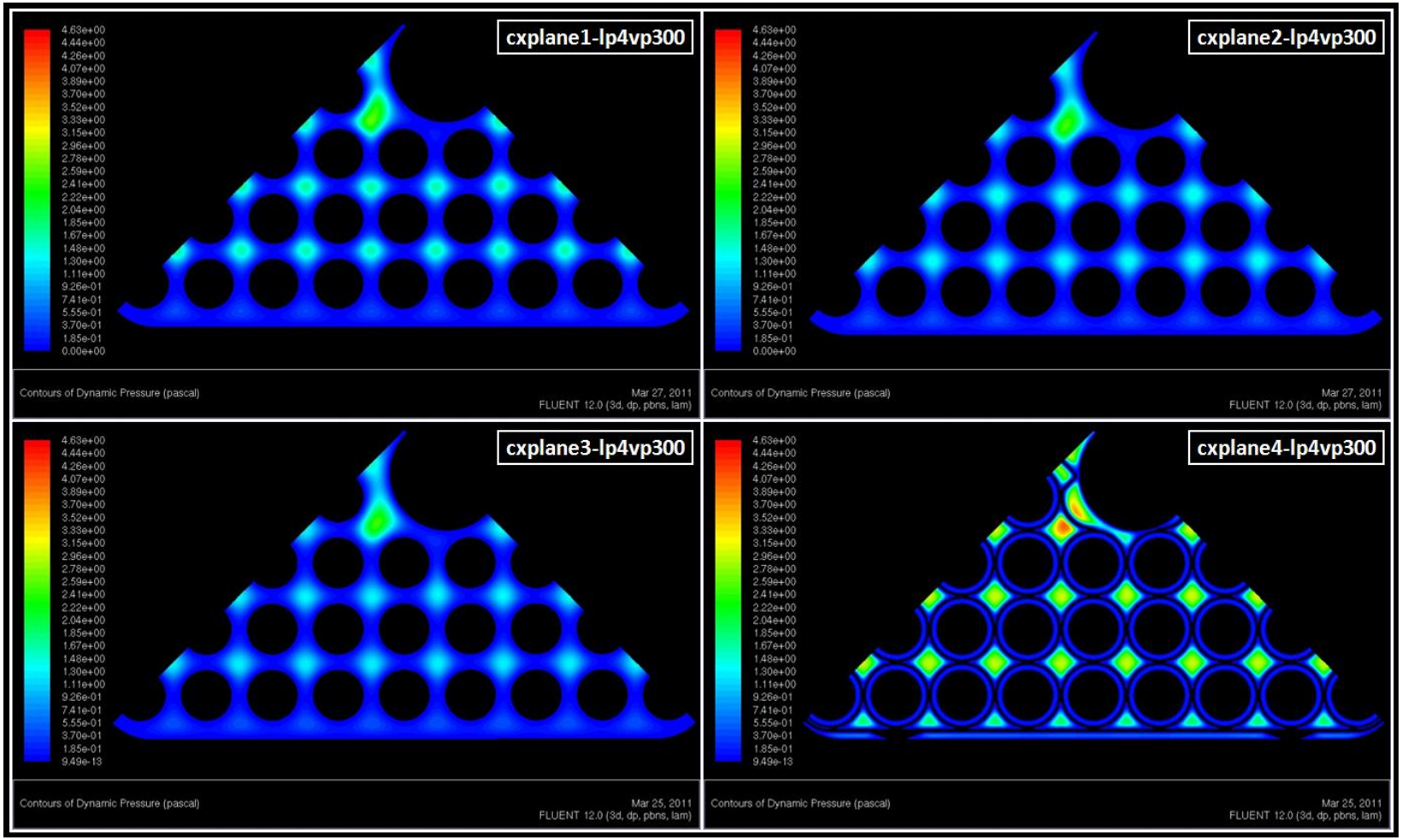


Figure 5.62: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 300 slpm (1 of 2).

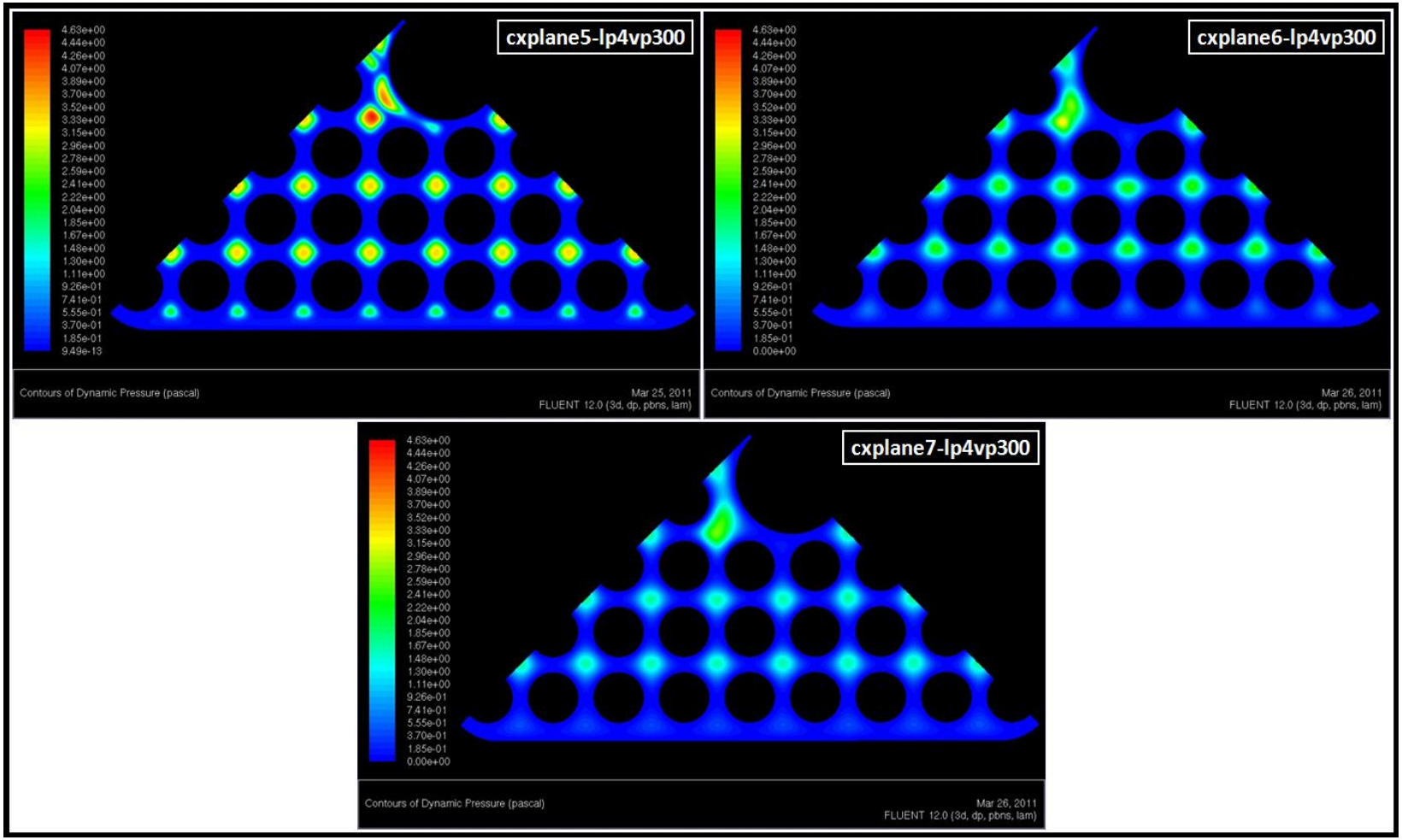


Figure 5.63: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 300 slpm (2 of 2).

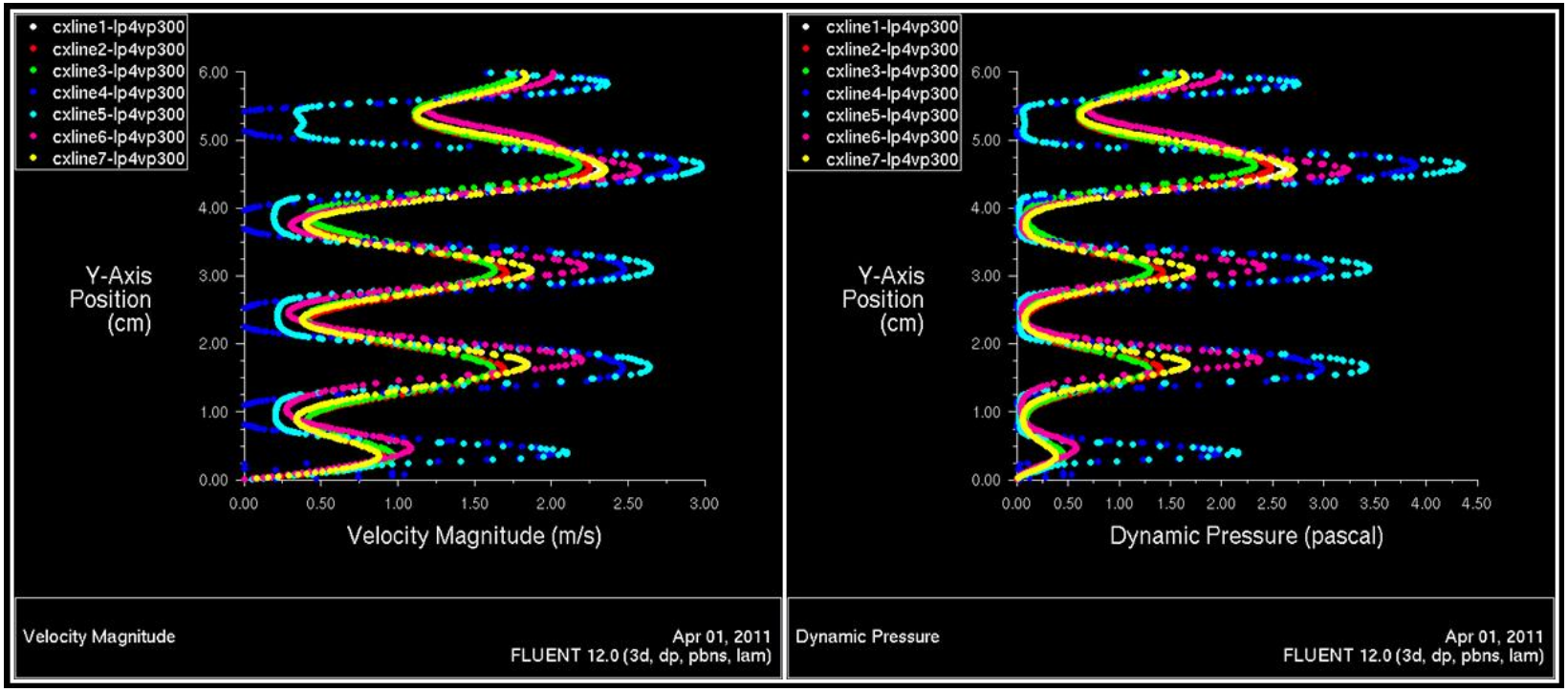


Figure 5.64: Velocity magnitude and dynamic pressure values for specified line surface profiles from loop 4 velocity trial at 300 slpm.

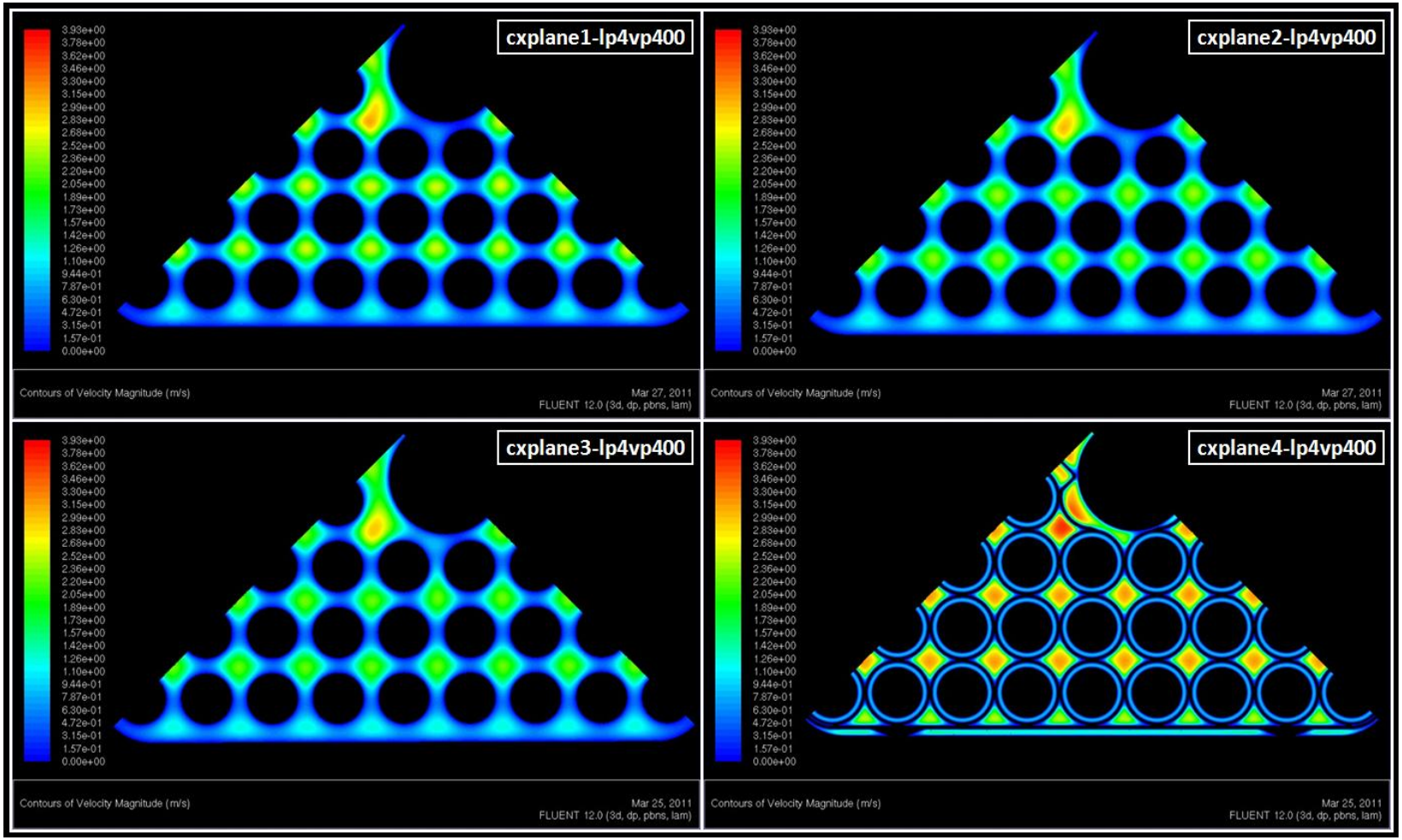


Figure 5.65: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 400 slpm (1 of 2).

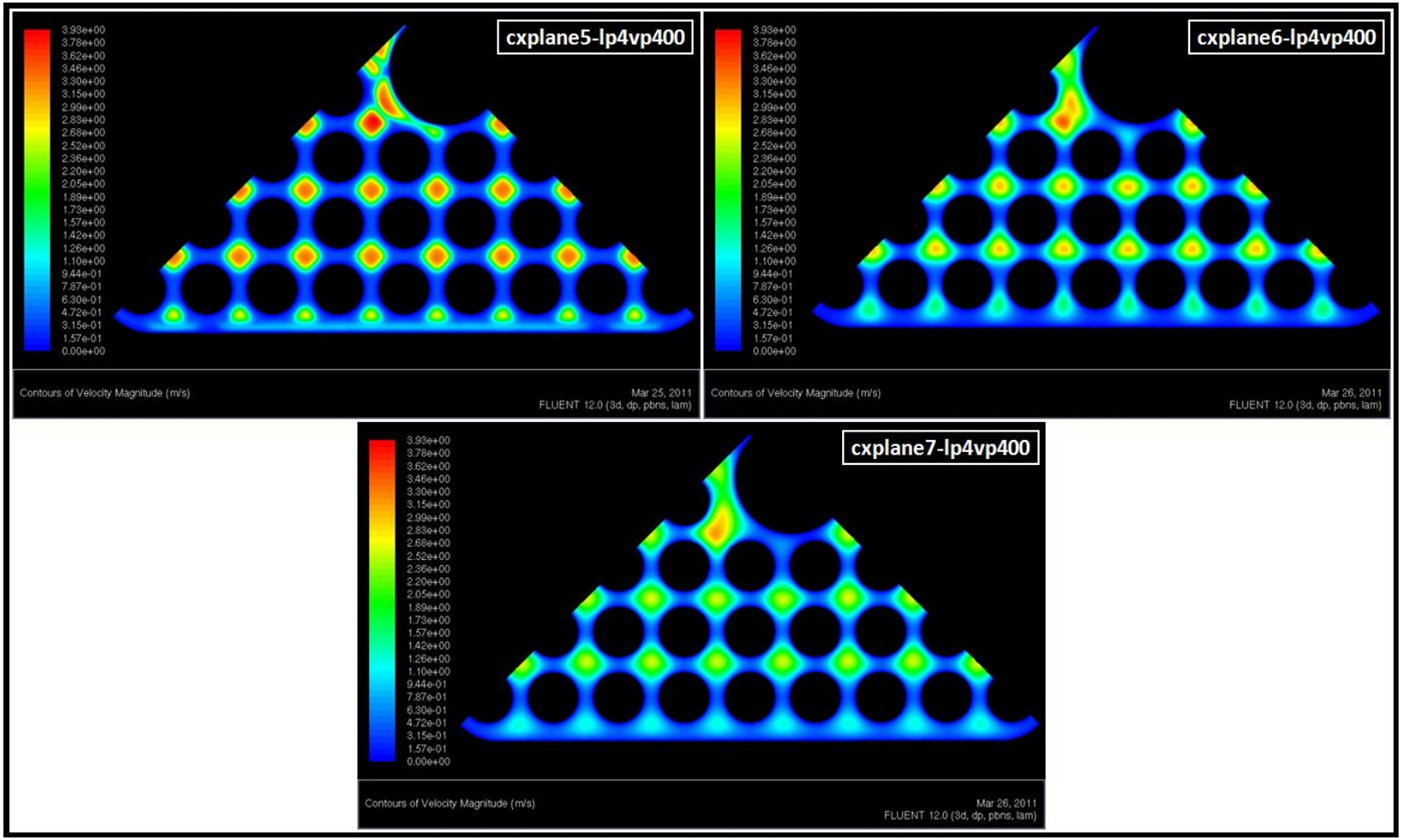


Figure 5.66: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 400 slpm (2 of 2).

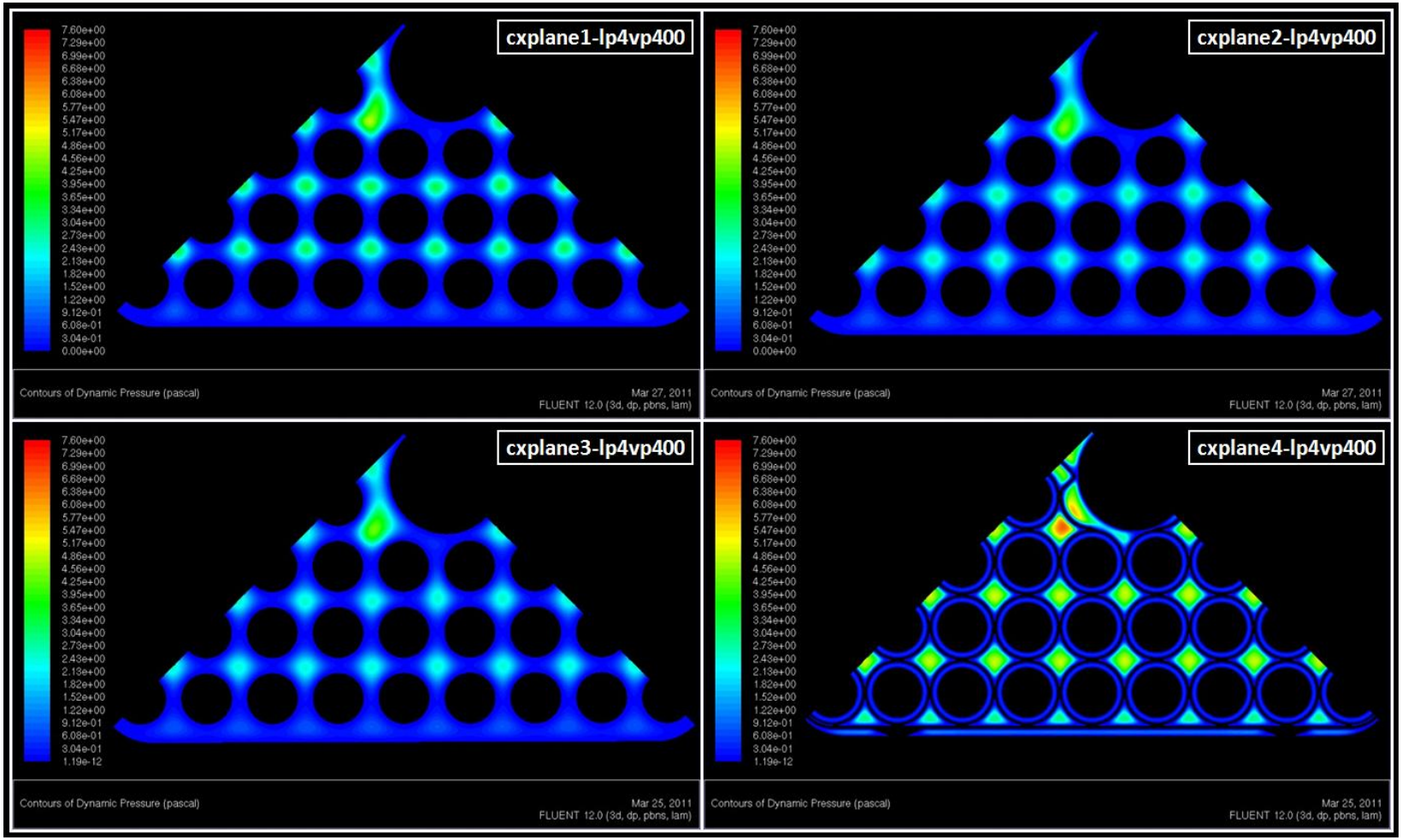


Figure 5.67: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 400 slpm (1 of 2).

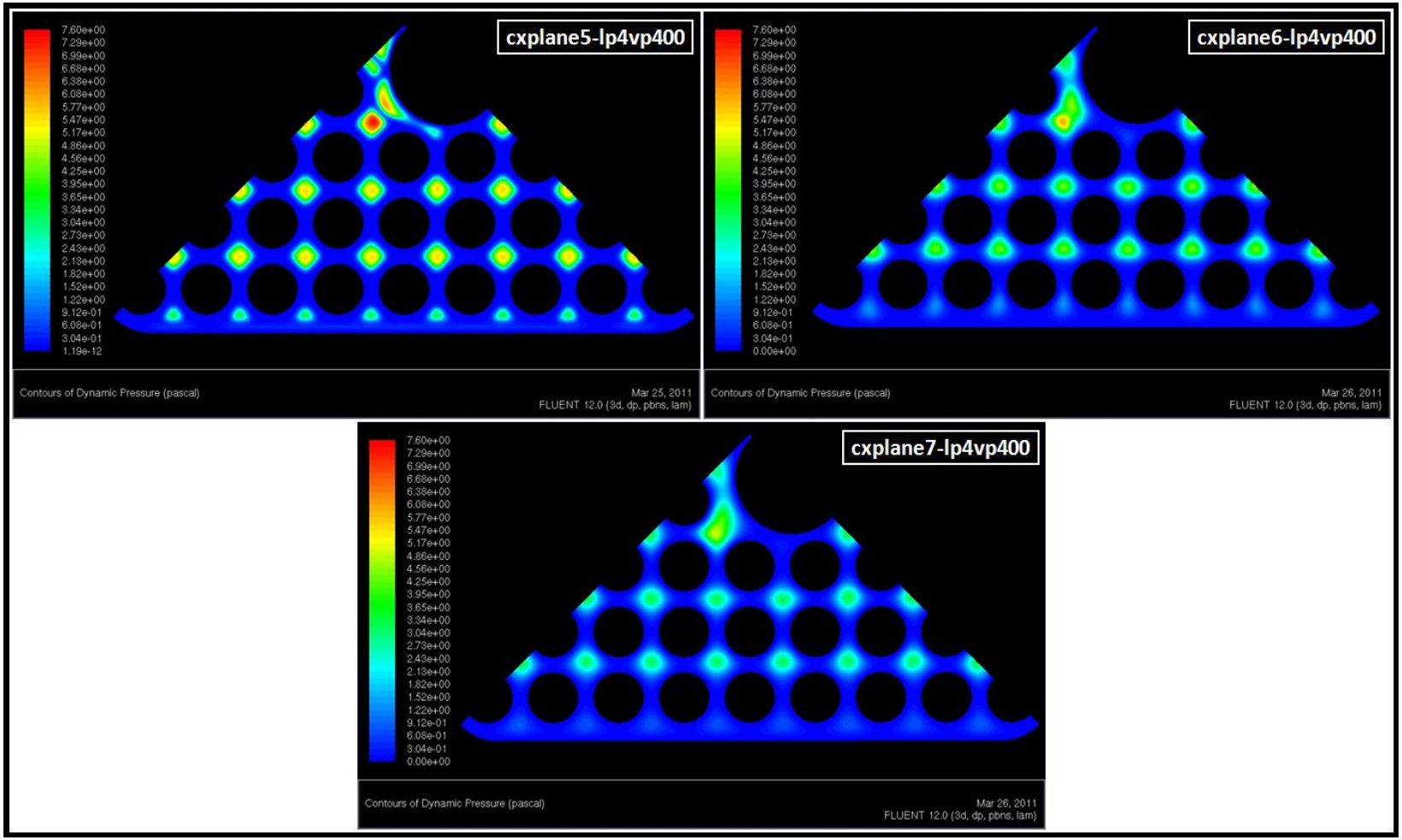


Figure 5.68: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 400 slpm (2 of 2).

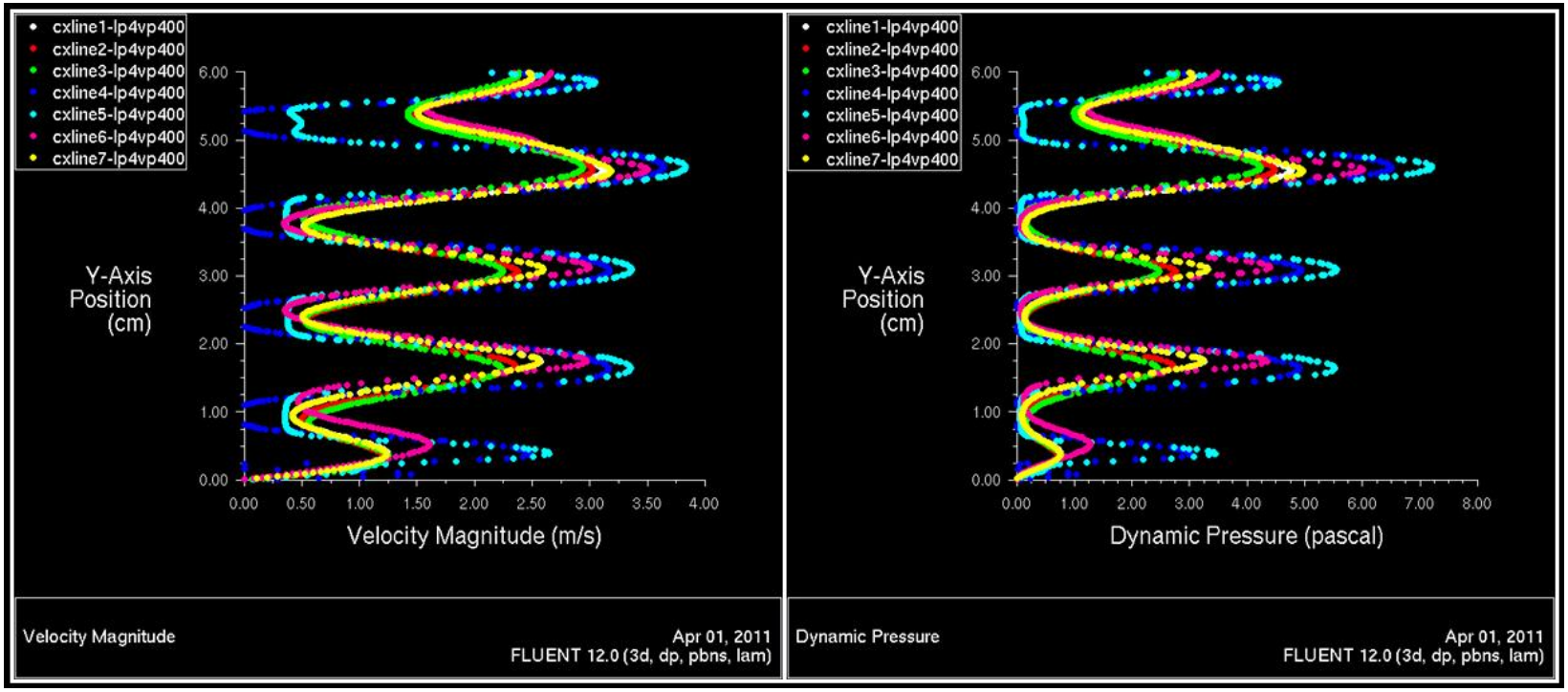


Figure 5.69: Velocity magnitude and dynamic pressure values for specified line surface profiles from loop 4 velocity trial at 400 slpm.

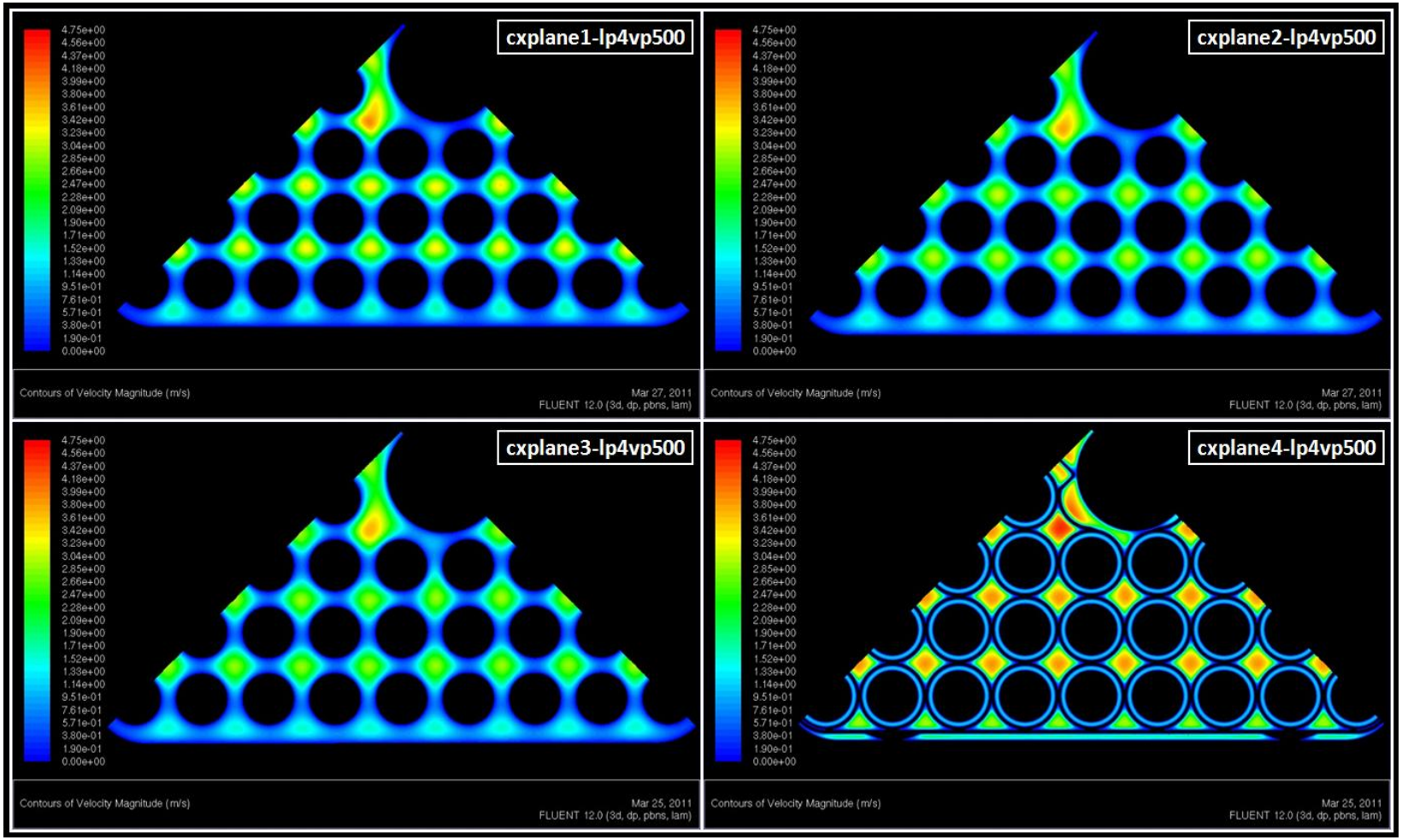


Figure 5.70: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 500 slpm (1 of 2).

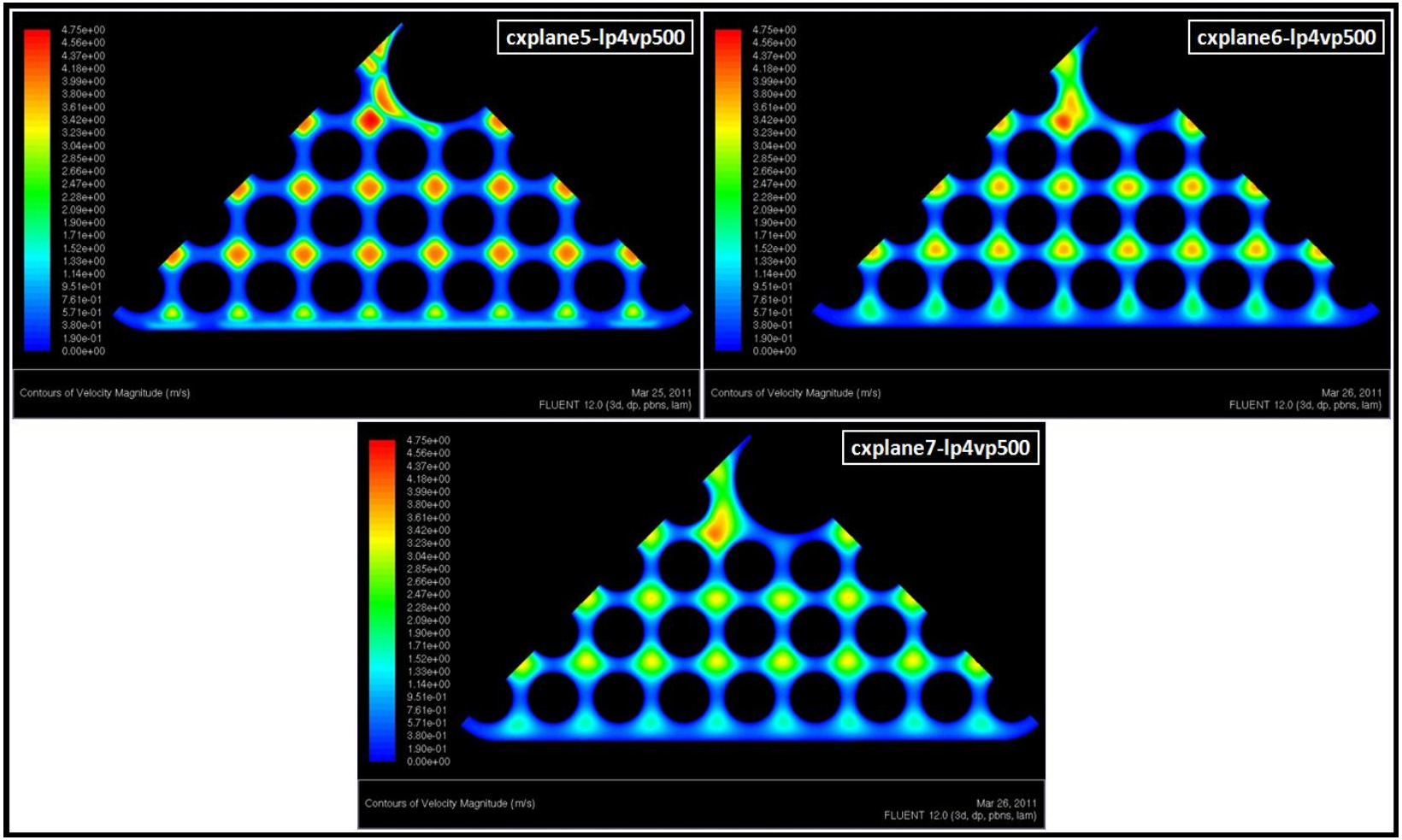


Figure 5.71: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 500 slpm (2 of 2).

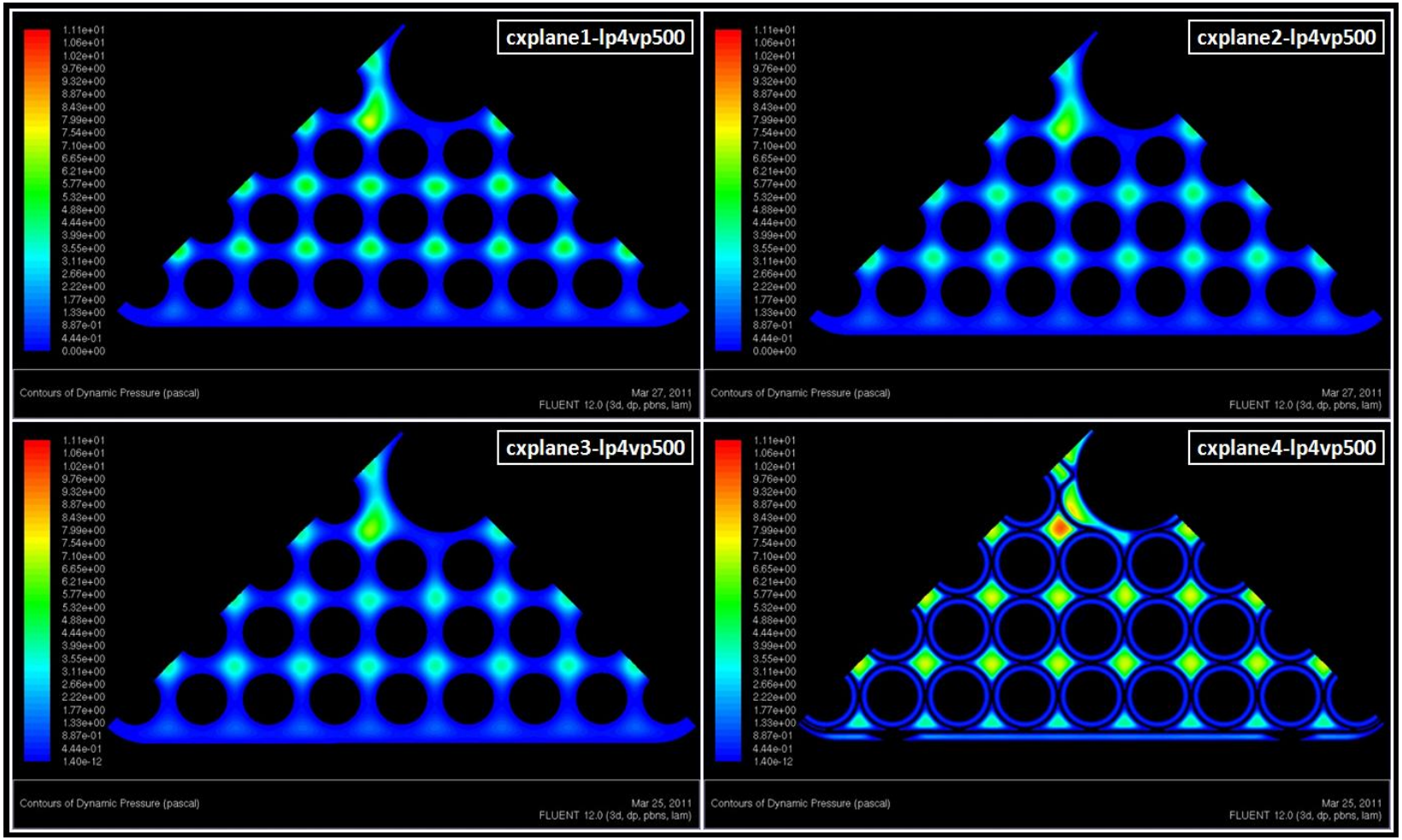


Figure 5.72: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 500 slpm (1 of 2).

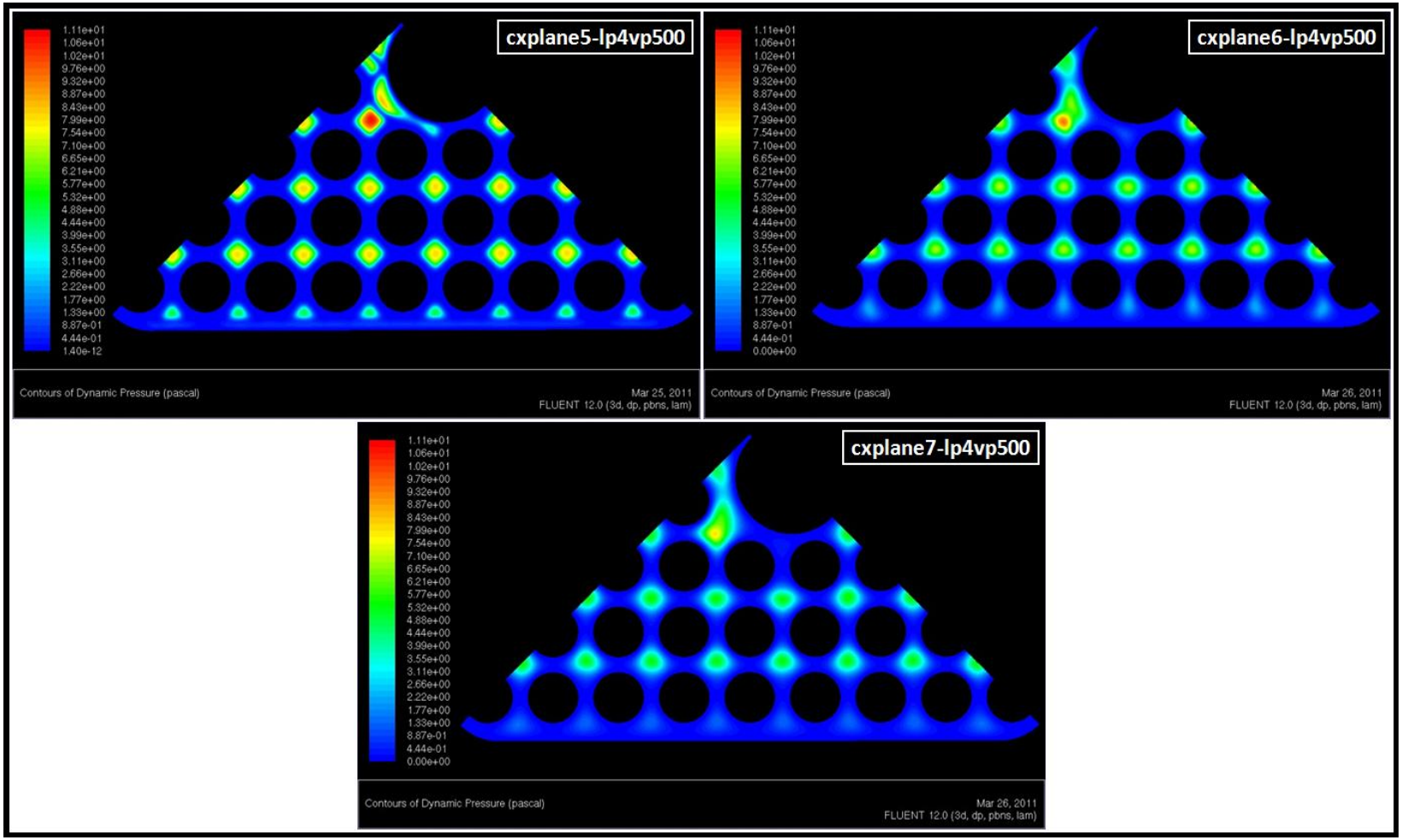


Figure 5.73: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 500 slpm (2 of 2).

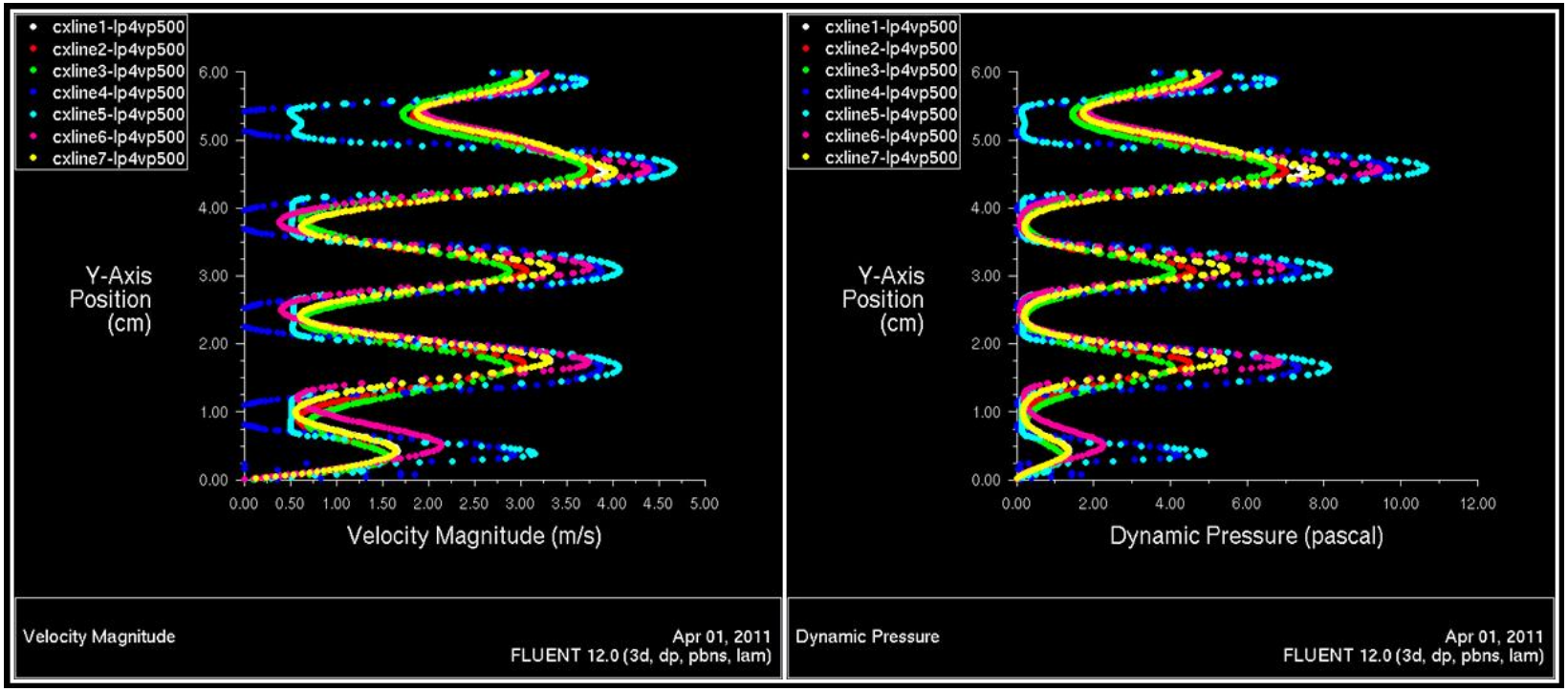


Figure 5.74: Velocity magnitude and dynamic pressure values for specified line surface profiles from loop 4 velocity trial at 500 slpm.

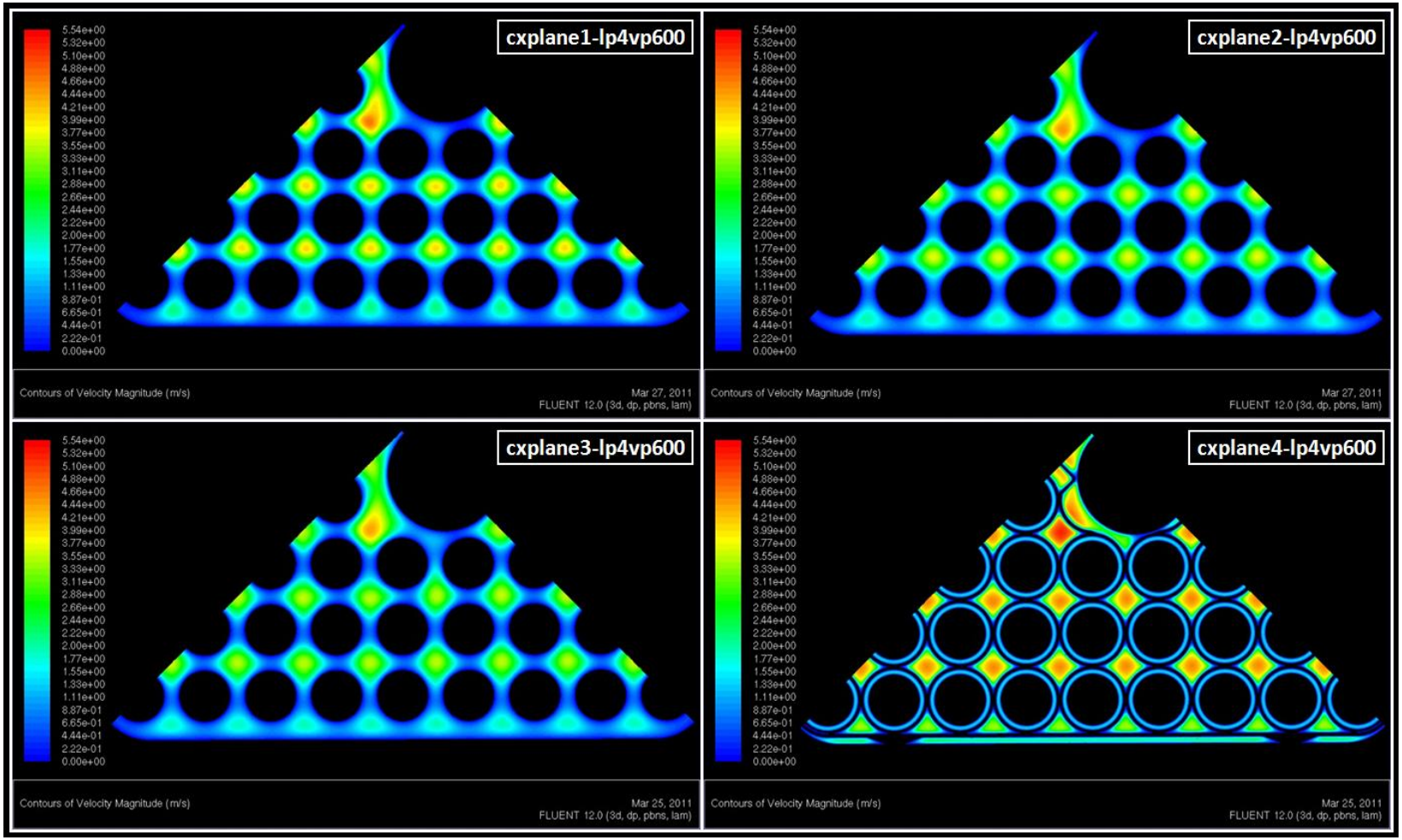


Figure 5.75: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 600 slpm (1 of 2).

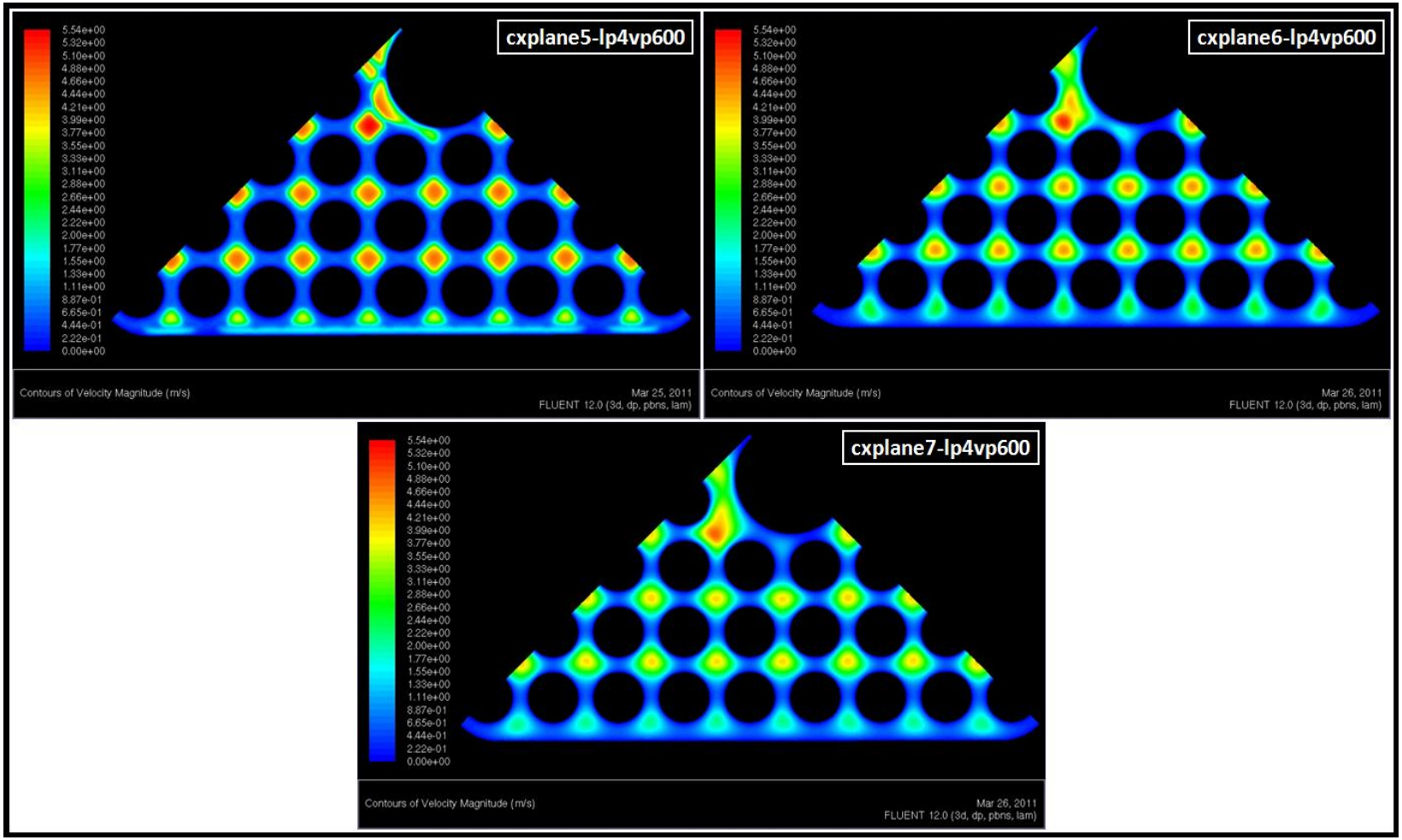


Figure 5.76: Contours of velocity magnitude across specified cross-sections for loop 4 velocity profile at 600 slpm (2 of 2).

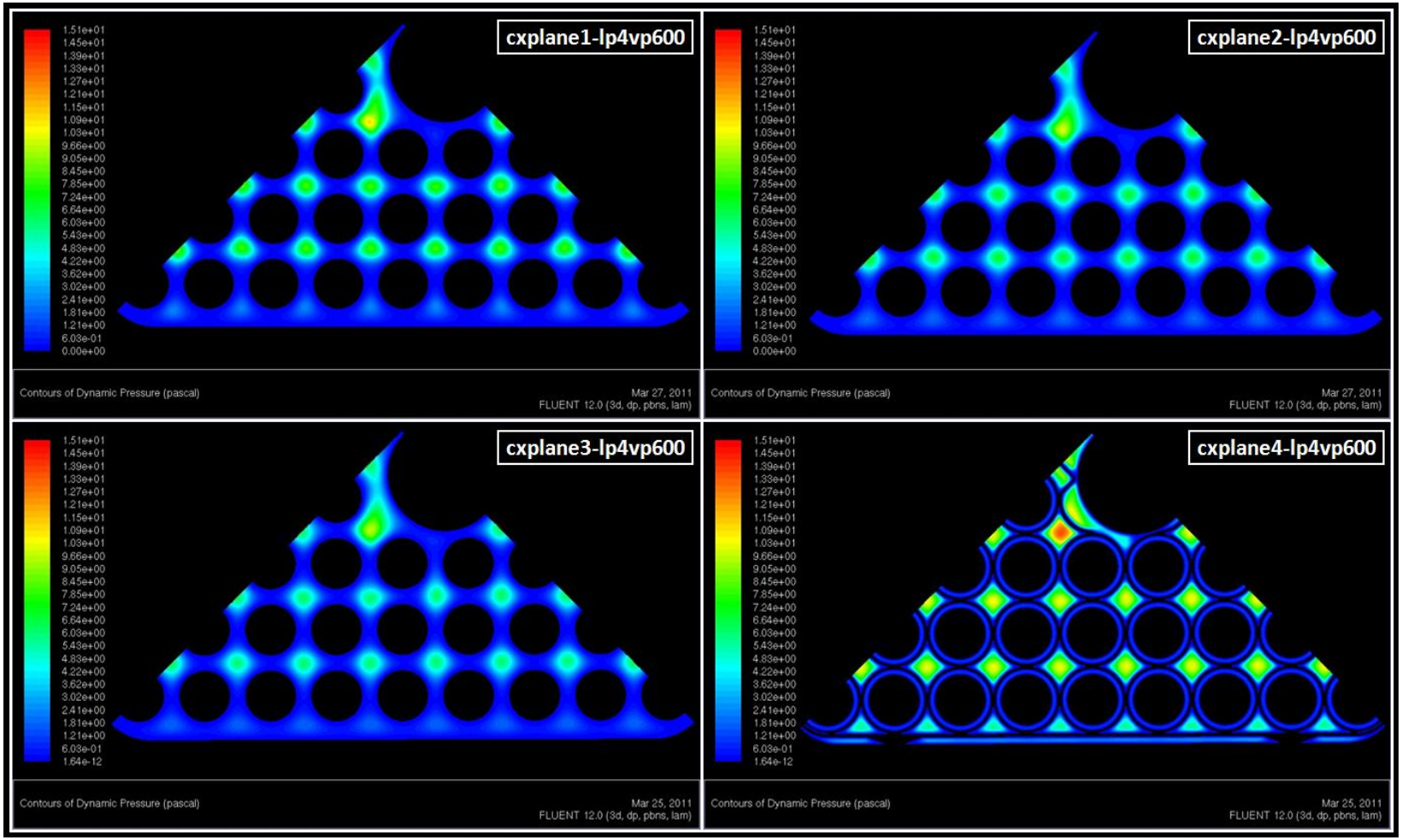


Figure 5.77: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 600 slpm (1 of 2).

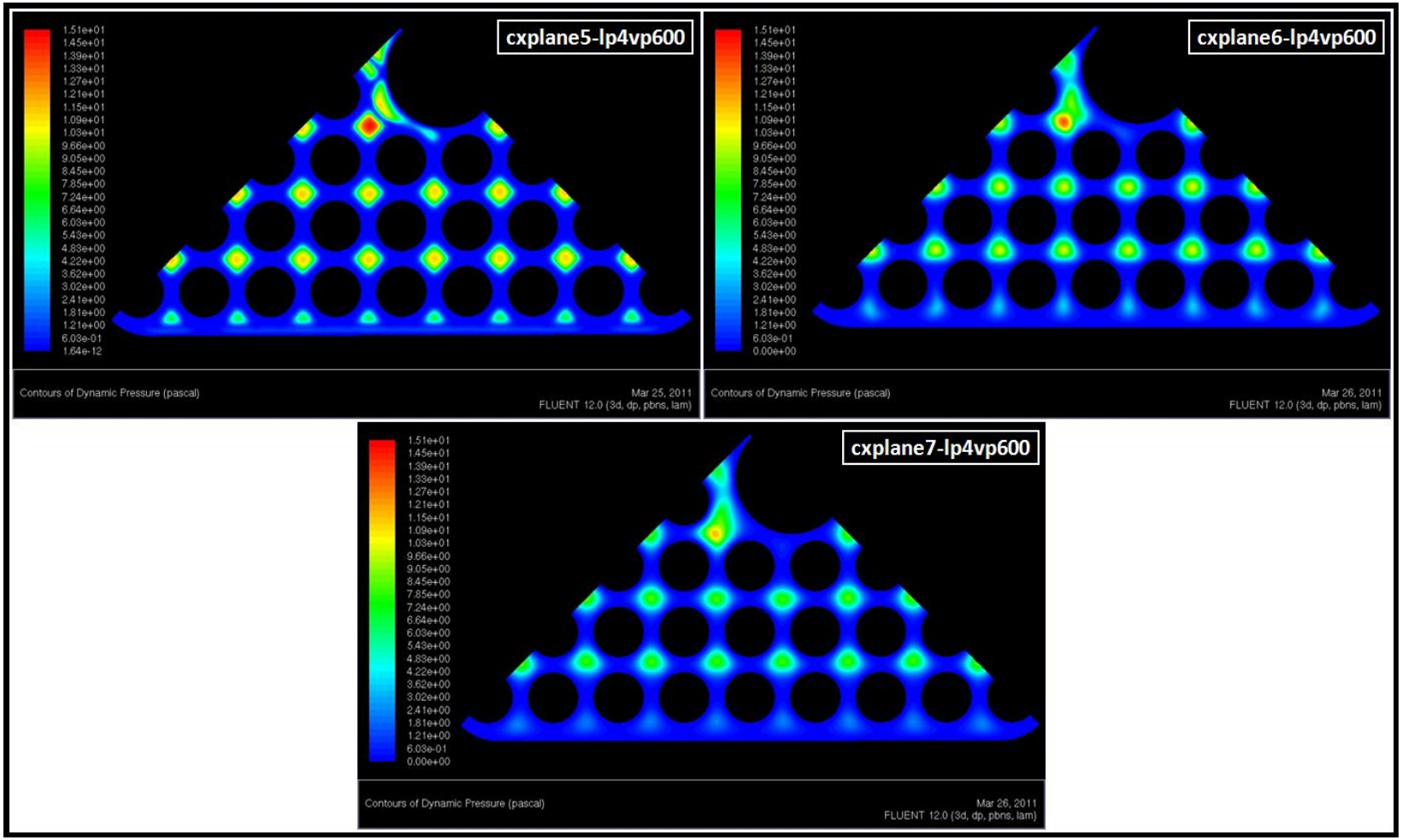


Figure 5.78: Contours of dynamic pressure across specified cross-sections for loop 4 velocity profile at 600 slpm (2 of 2).

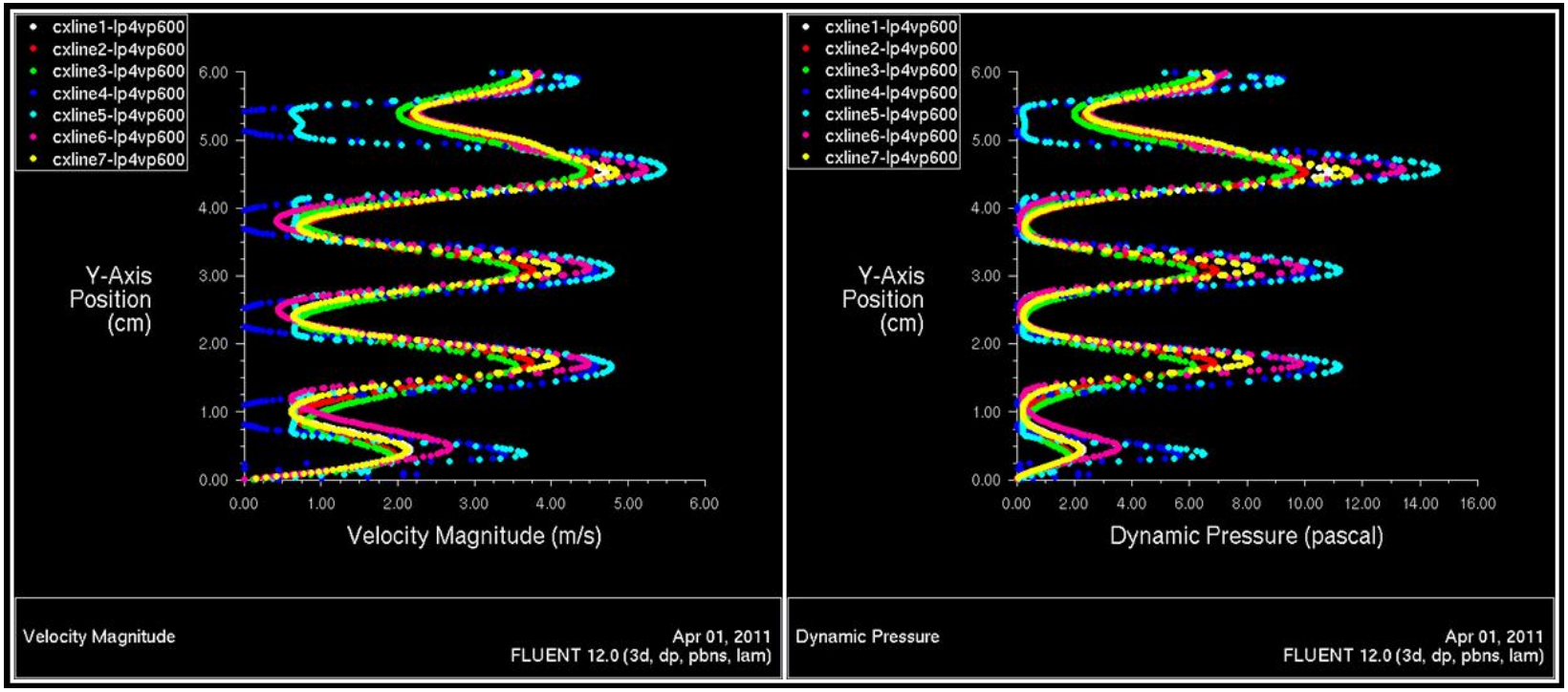


Figure 5.79: Velocity magnitude and dynamic pressure values for specified line surface profiles from loop 4 velocity trial at 600 slpm.

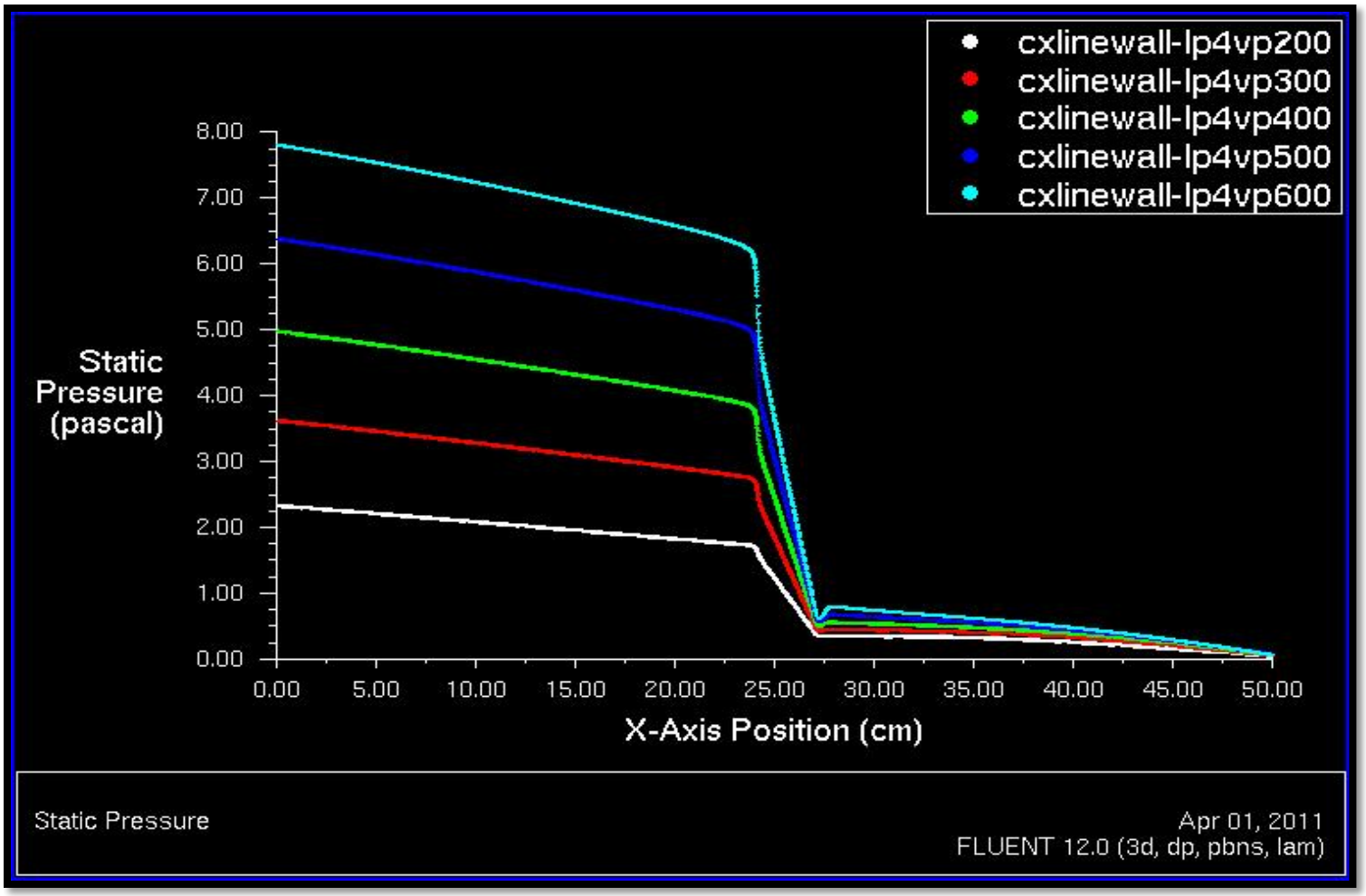


Figure 5.80: Static pressures from 'cxlinewall' line surface along model length at the canister wall for all loop 4 velocity trials.

The graphical results shown in Figures 5.55 through 5.80 for the final loop 4 velocity simulations at five flow rates spanning 200 to 600 slpm afford a valuable insight into the desired characterization of developing flow-field solutions across domain segments involving both fully-populated fuel bundle and spacer assembly components. Although these images provide support for a number of different possible formulations and inferences concerning typical flow fields within a fuel assembly structure, a few points of notable interest warrant particular focus at the present stage of research development. Foremost, the contour as well as line-surface profile plots at the inlet (*i.e.* cxplane1/cxline1-lp4vp#) and outlet (*i.e.* cxplane7/cxline7-lp4vp#) of the modeled domain are nearly identical at each flow rate compared for both velocity magnitude and dynamic pressure parameters, which suggests the establishment of reasonably periodic flow-field solutions. This substantiates the initial criteria implemented for judging adequate flow periodicity development after a certain number of velocity profile loops as a less than 1% difference in both MWA total and AWA static pressure differentials between successive loop trials over the entire model at all airflow rates.

Furthermore, a similar comparative analysis between contour as well as line-surface profile plots 1 cm upstream (*i.e.* cxplane3/cxline3-lp4vp#) and 1 cm downstream (*i.e.* cxplane5/cxline5-lp4vp#) of the spacer component reveals significantly dissimilar flow-field values at each flow rate for both the velocity magnitude and dynamic pressure solution parameters. Again, this provides some level of credibility toward the previously expressed concerns regarding necessary and accurate accounting of the difference between kinetic energy coefficients at pressure measurement locations of varying flow-field development; these are discussed in accordance with Equations (5.30) through (5.32) above. The importance of selecting specific pressure port locations in the near upstream and downstream vicinities of a spacer component is further compounded by the

substantial static pressure gradient(s) present at these localities as illustrated in Figure 5.80 for measurements along the axial model length just offset from the canister wall, especially the static pressure fluctuations shown immediately downstream.

In addition to the preceding, postprocessed graphics presented above representing flow-field solutions from each of the loop 4 velocity simulations, a modest effort is expended to initially assess the capabilities and associated resource costs of generating some of the highest-end, three-dimensional, postprocessed graphics in the FLUENT CFD software using extensive scene overlay features available within the latest distribution. Ultimately, almost every image in this set necessitated a considerable segment of dedicated wall-clock time to produce with several images requiring between two and three hours for completion of the final scene overlay. Moreover, in order to maintain a reasonable memory requirement ranging between 128 and 256 GB of RAM for image postprocessing, the modeled domain is divided into three separate flow volumes and the corresponding solution data files from a designated airflow rate are interpolated onto each related volume grid section for independent consideration (as defined in Figure 5.81 below). Therefore, the following high-end, postprocessed graphics shown in ensuing Figures 5.82 through 5.103 are limited to the loop 4 velocity trial solutions at the lowest and highest flow rates examined (*i.e.* 200 and 600 slpm, respectively). These results impart additional clarification and evidence in support of the aforementioned discussion points of noted significance.

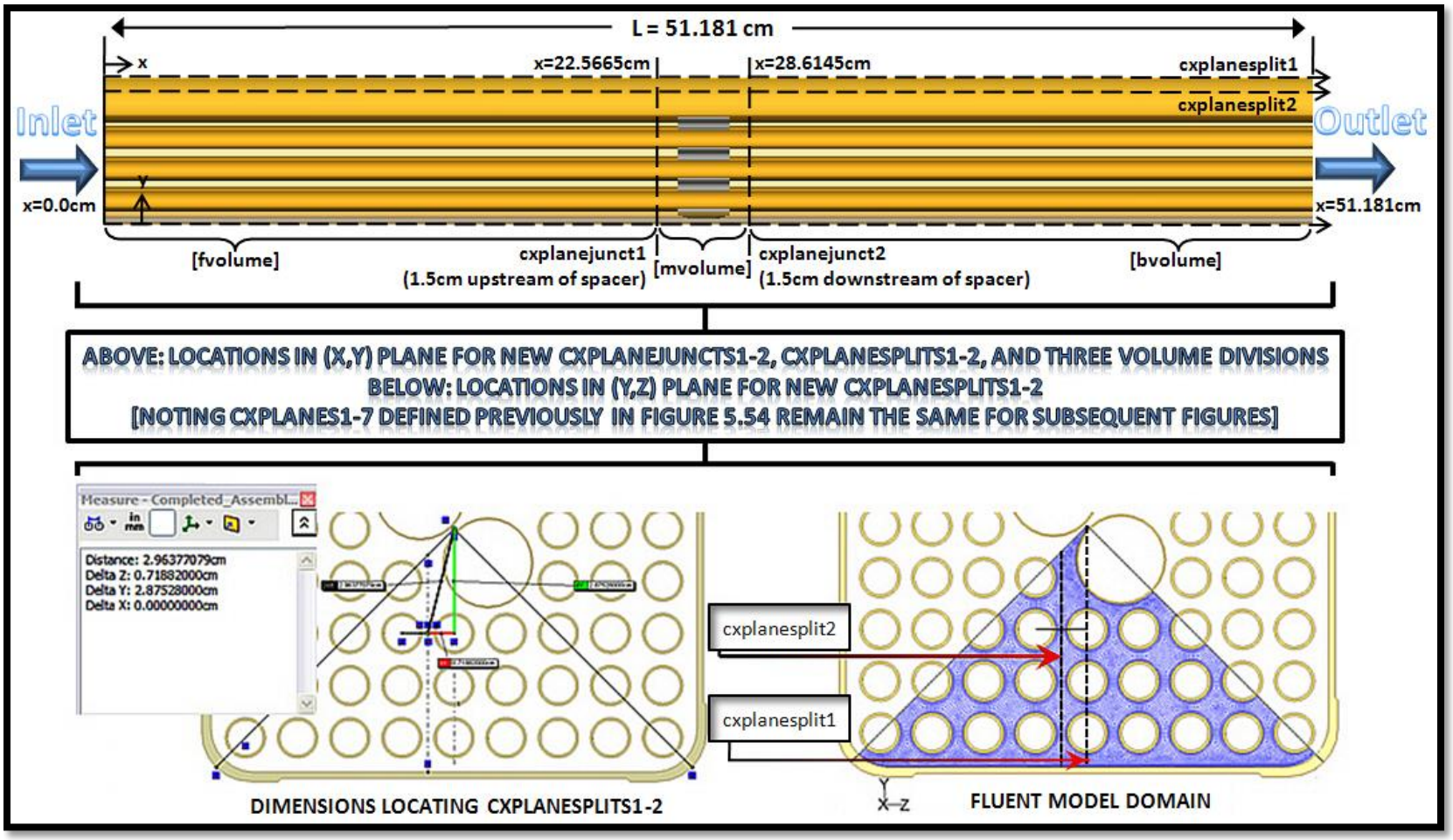


Figure 5.81: Defined locations of new cross-sectional planes and volume divisions, which are referenced throughout the following postprocessed graphics in addition to the cross-sectional planes defined previously in Figure 5.54.

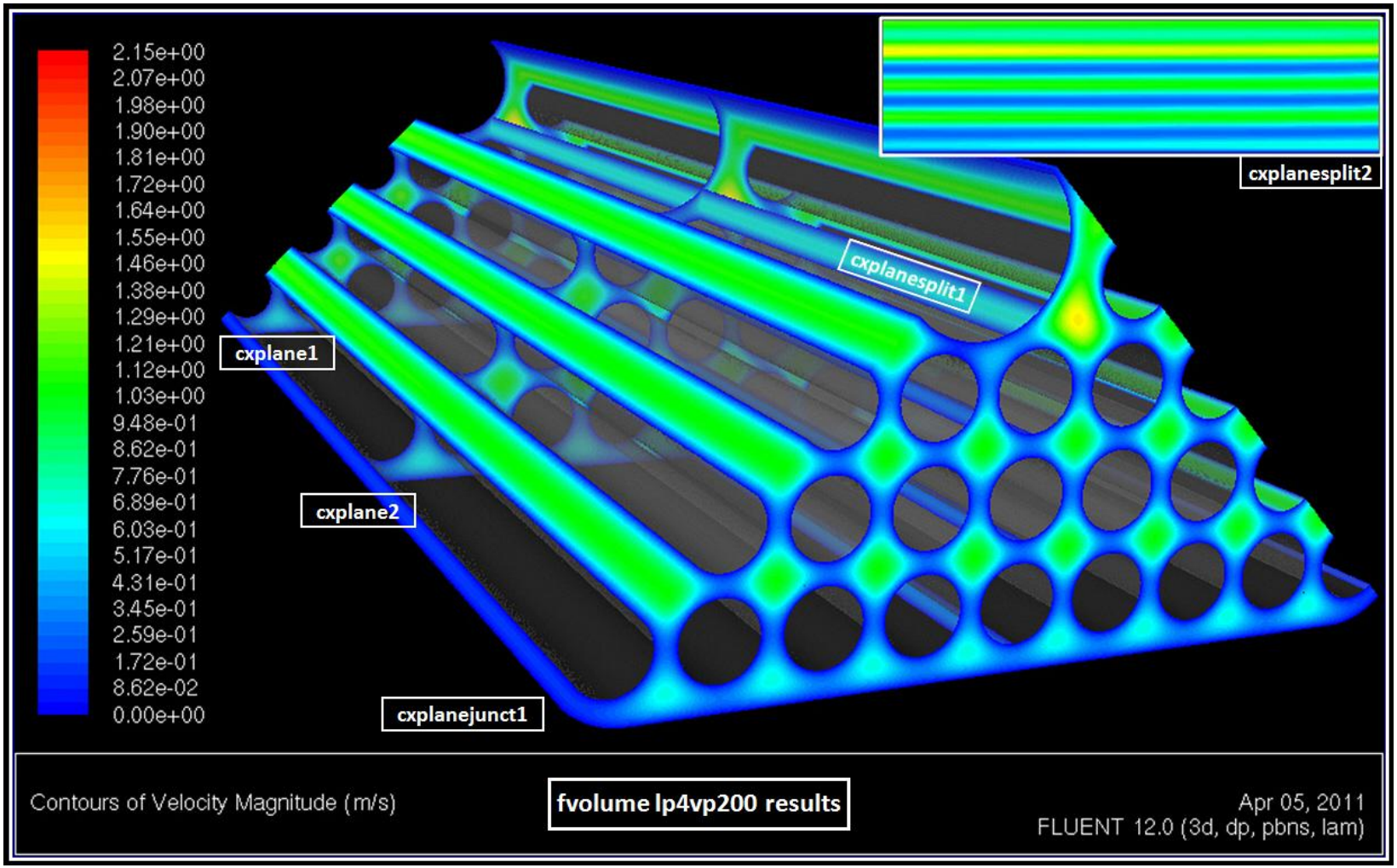


Figure 5.82: Contours of velocity magnitude over specified cross-sections of fvolume for the loop 4 velocity trial at 200 slpm.

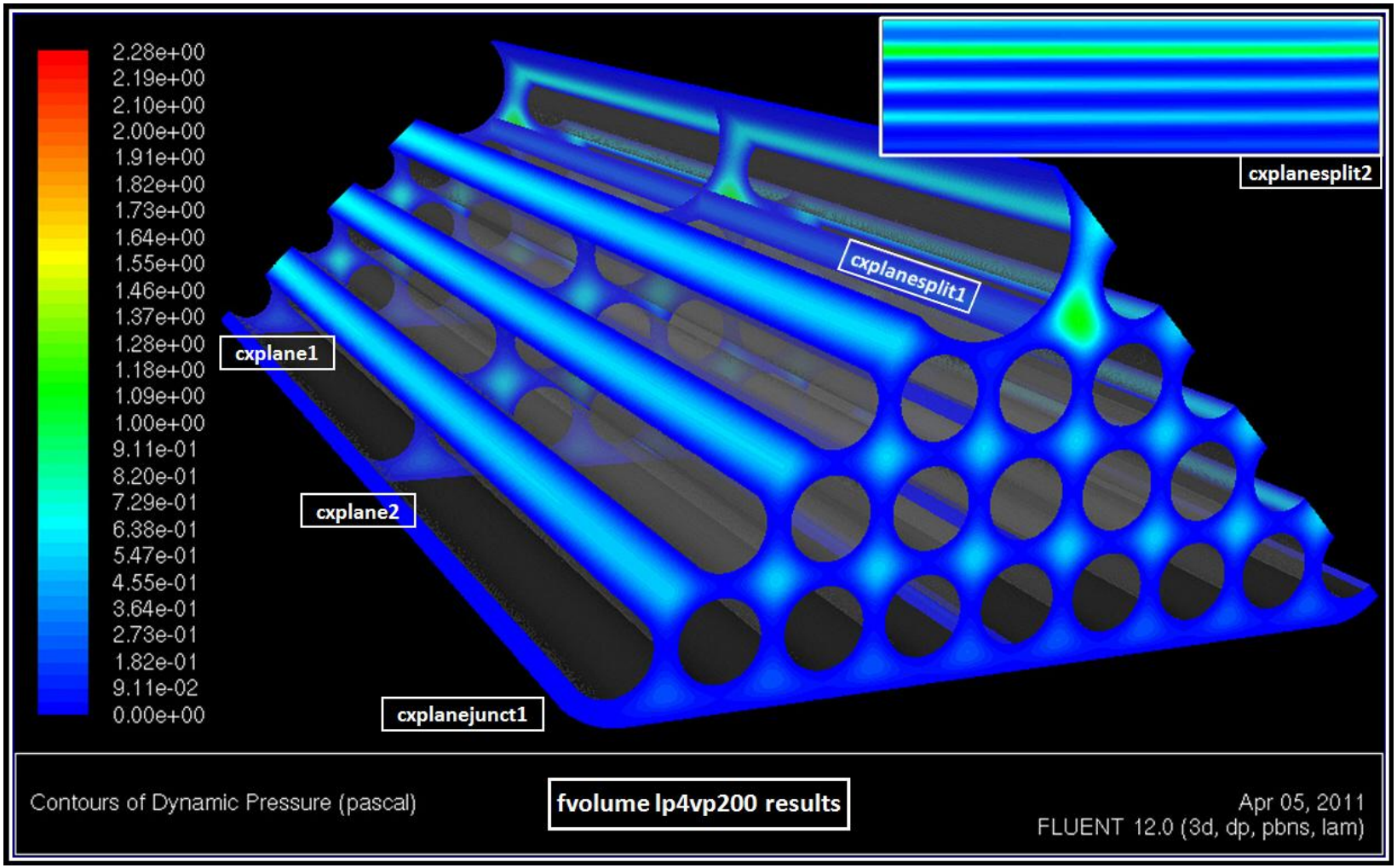


Figure 5.83: Contours of dynamic pressure over specified cross-sections of fvolume for the loop 4 velocity trial at 200 slpm.

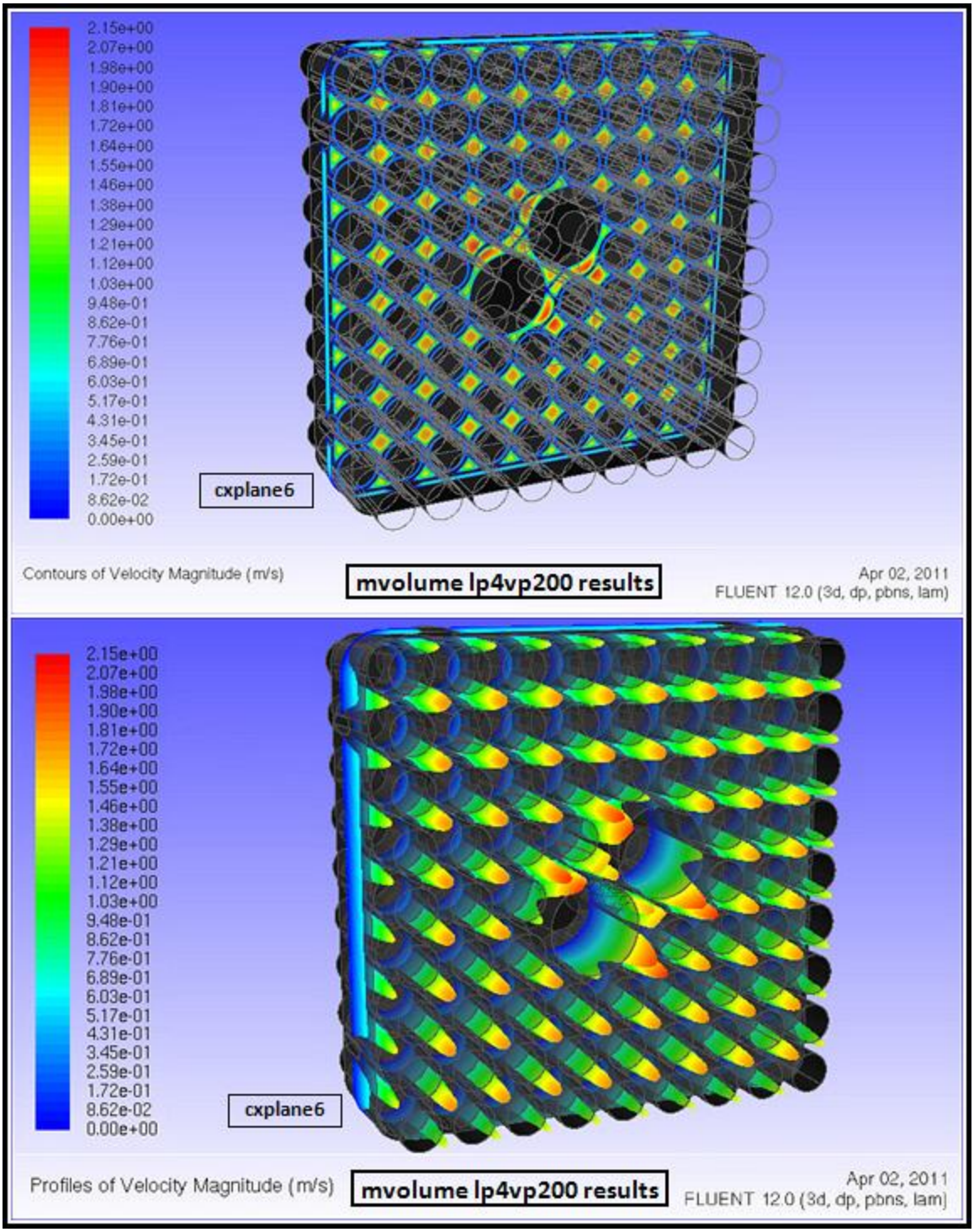


Figure 5.84: Contours and profiles of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm without symmetry.

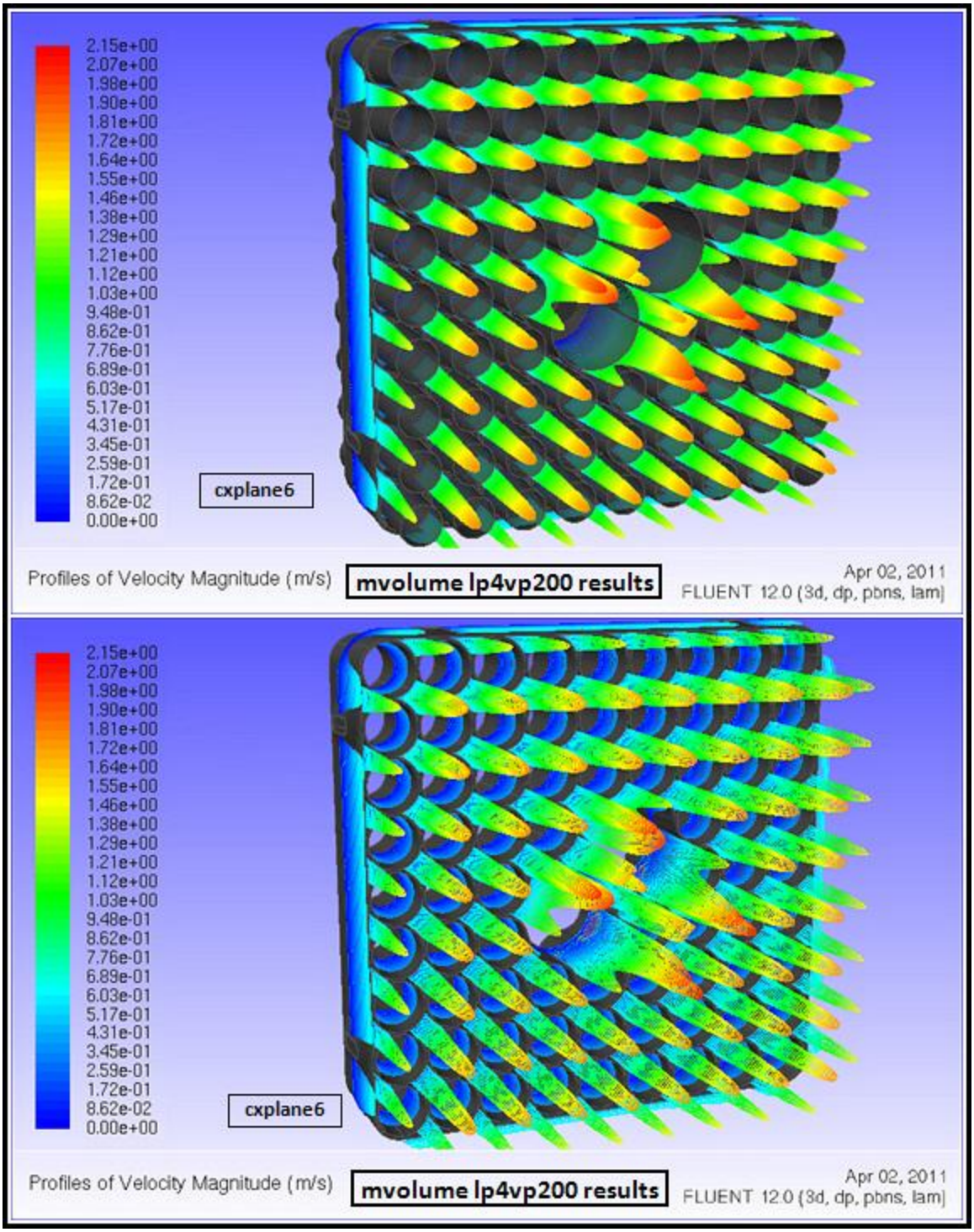


Figure 5.85: Profiles of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm without symmetry.

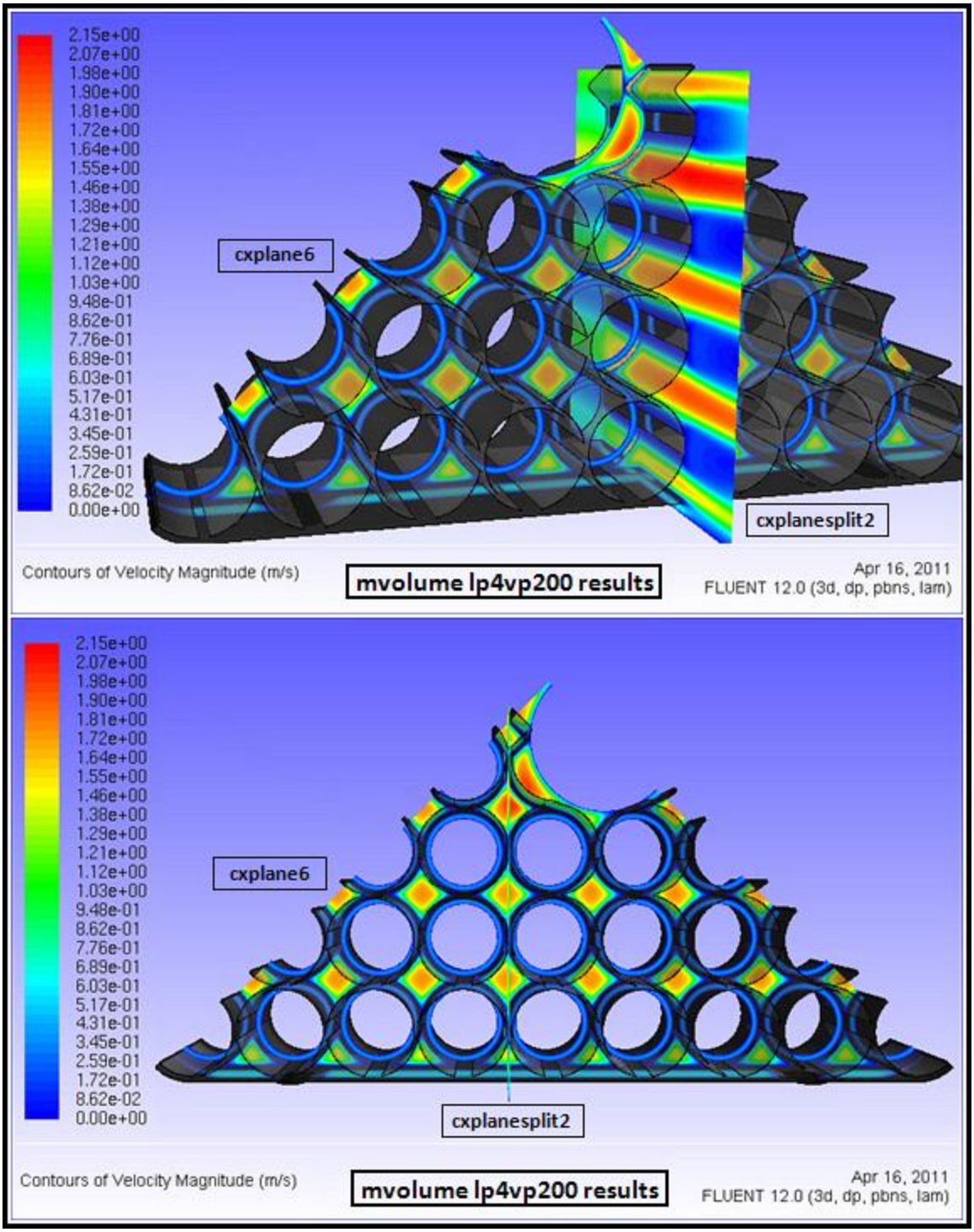


Figure 5.86: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm (1 of 3).

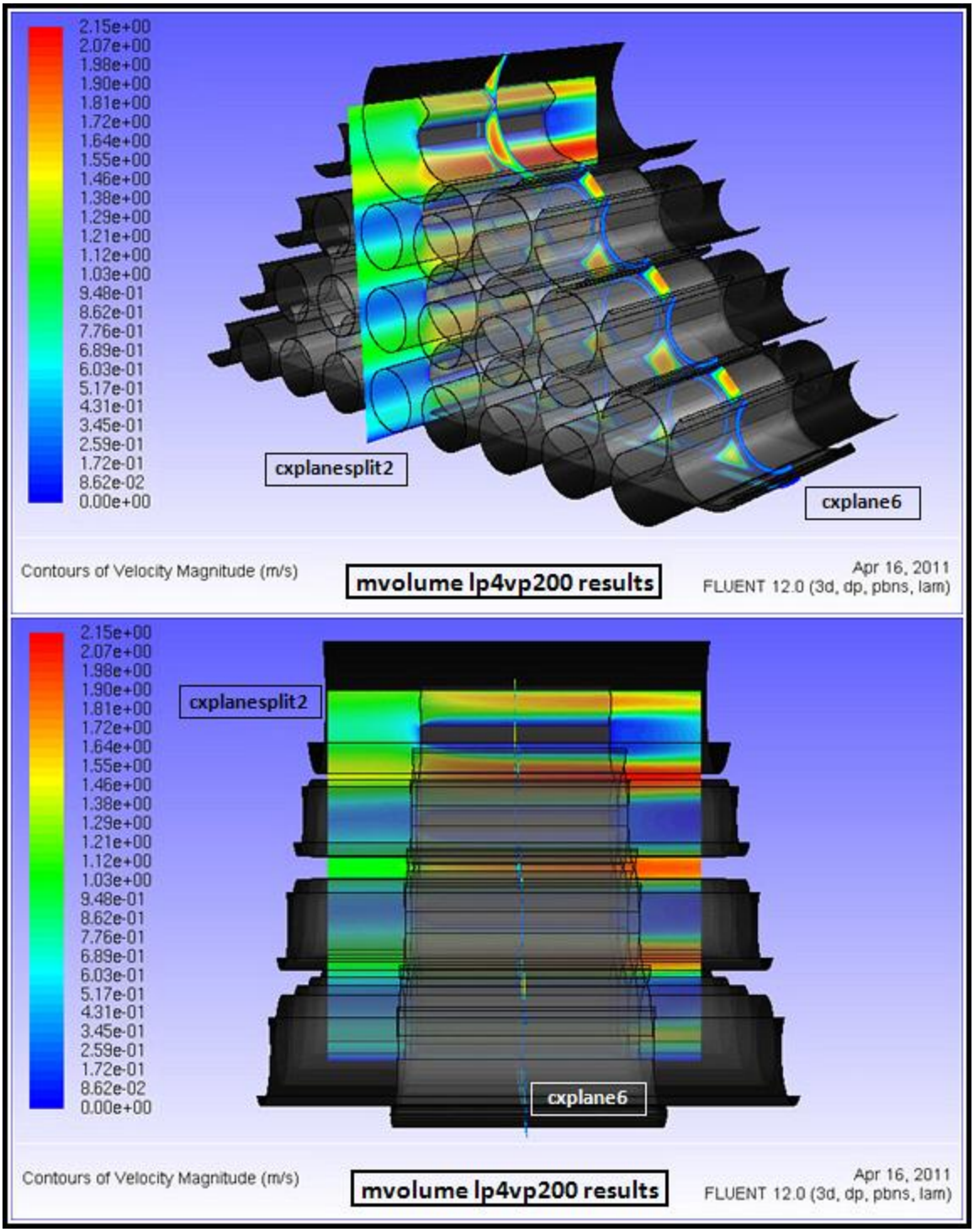


Figure 5.87: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm (2 of 3).

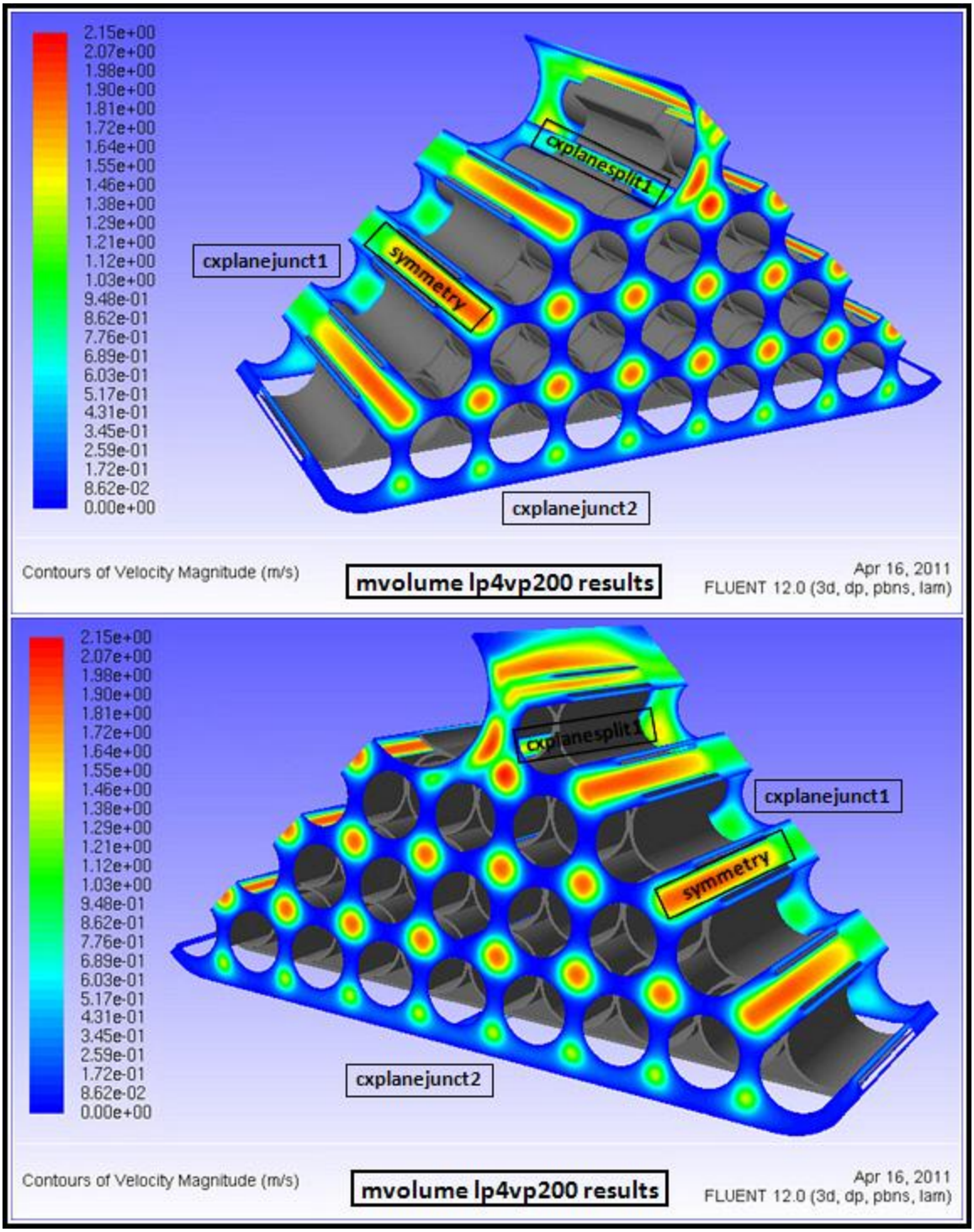


Figure 5.88: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm (3 of 3).

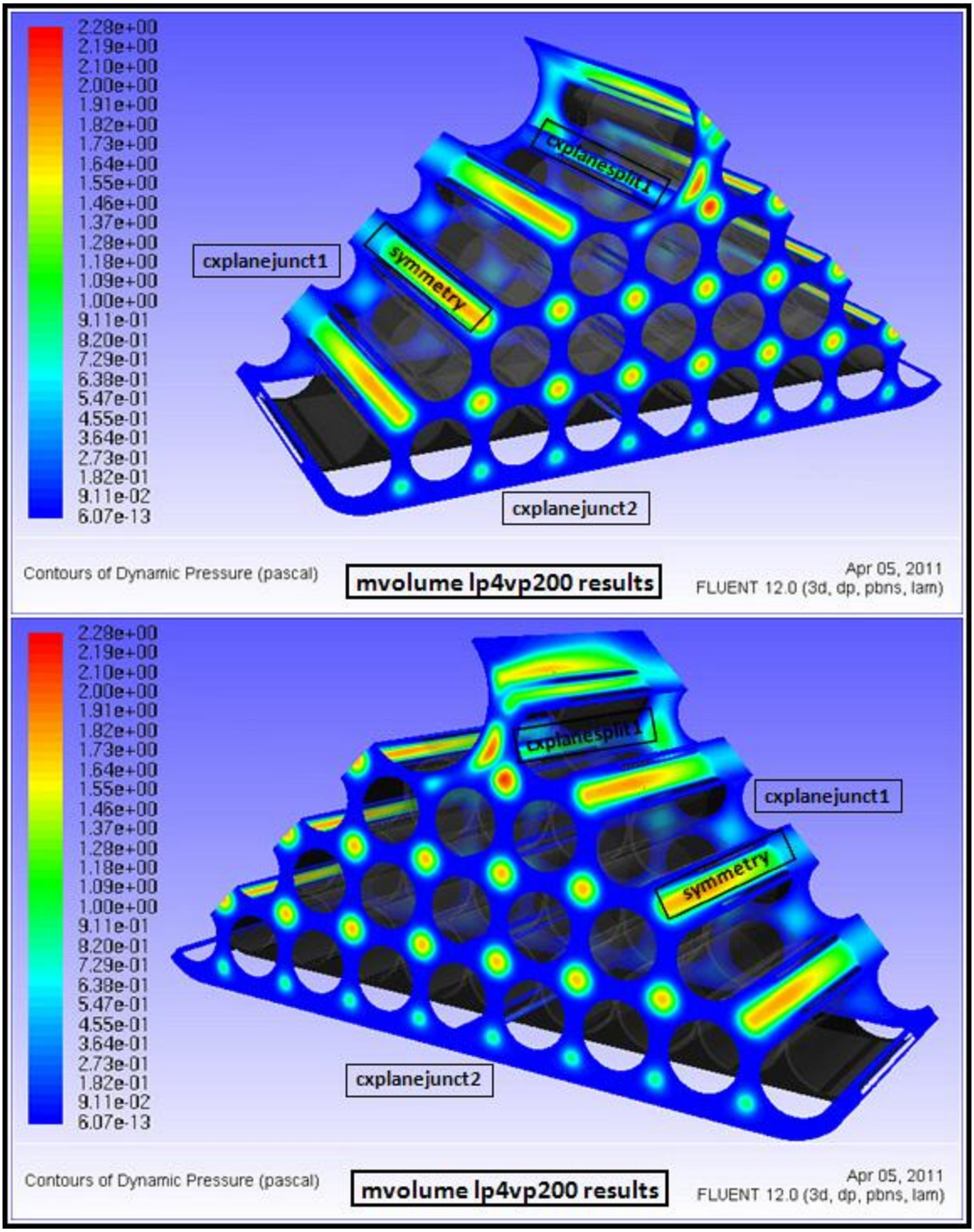


Figure 5.89: Contours of dynamic pressure over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm (1 of 2).

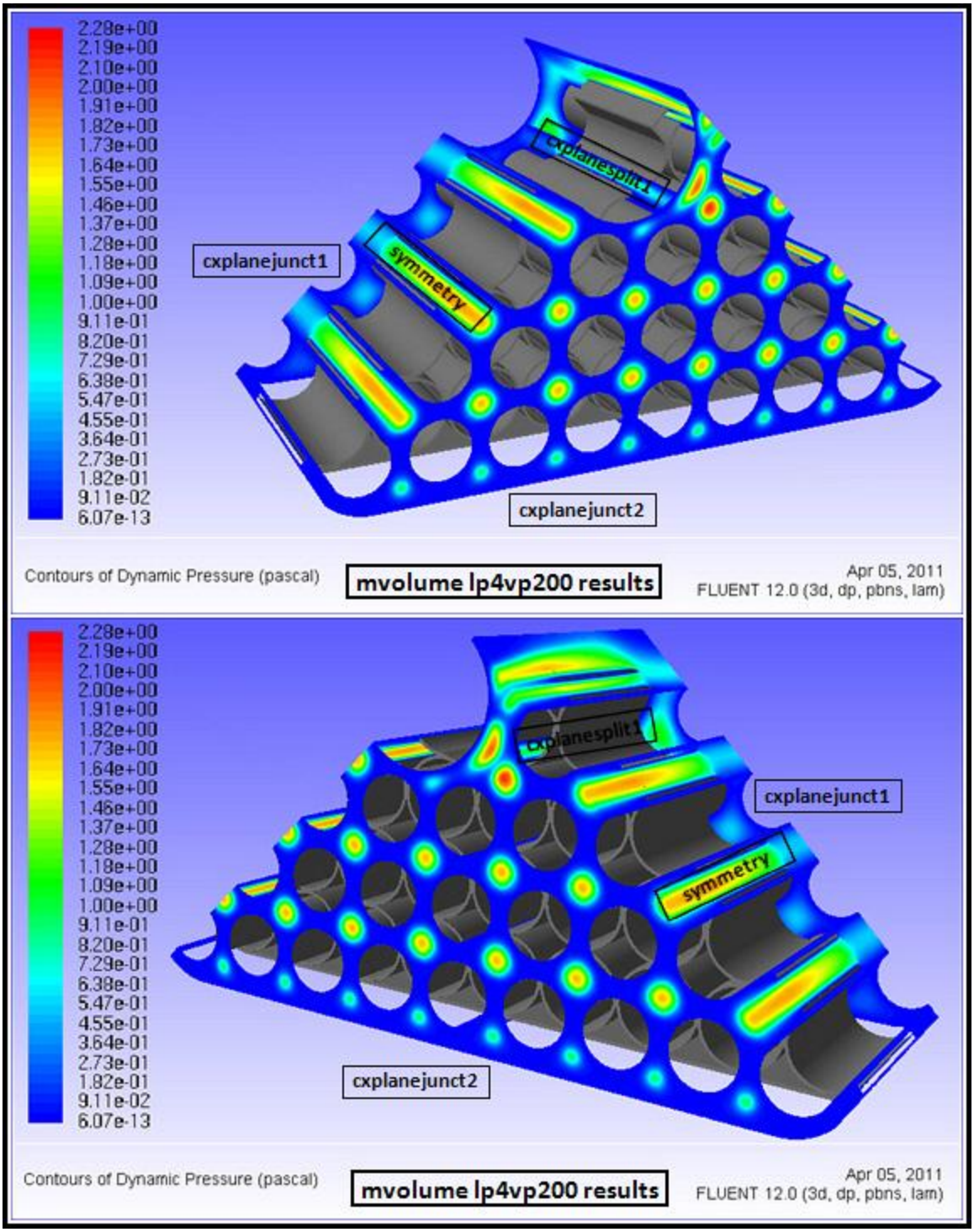


Figure 5.90: Contours of dynamic pressure over specified cross-sections of mvolume for the loop 4 velocity trial at 200 slpm (2 of 2).

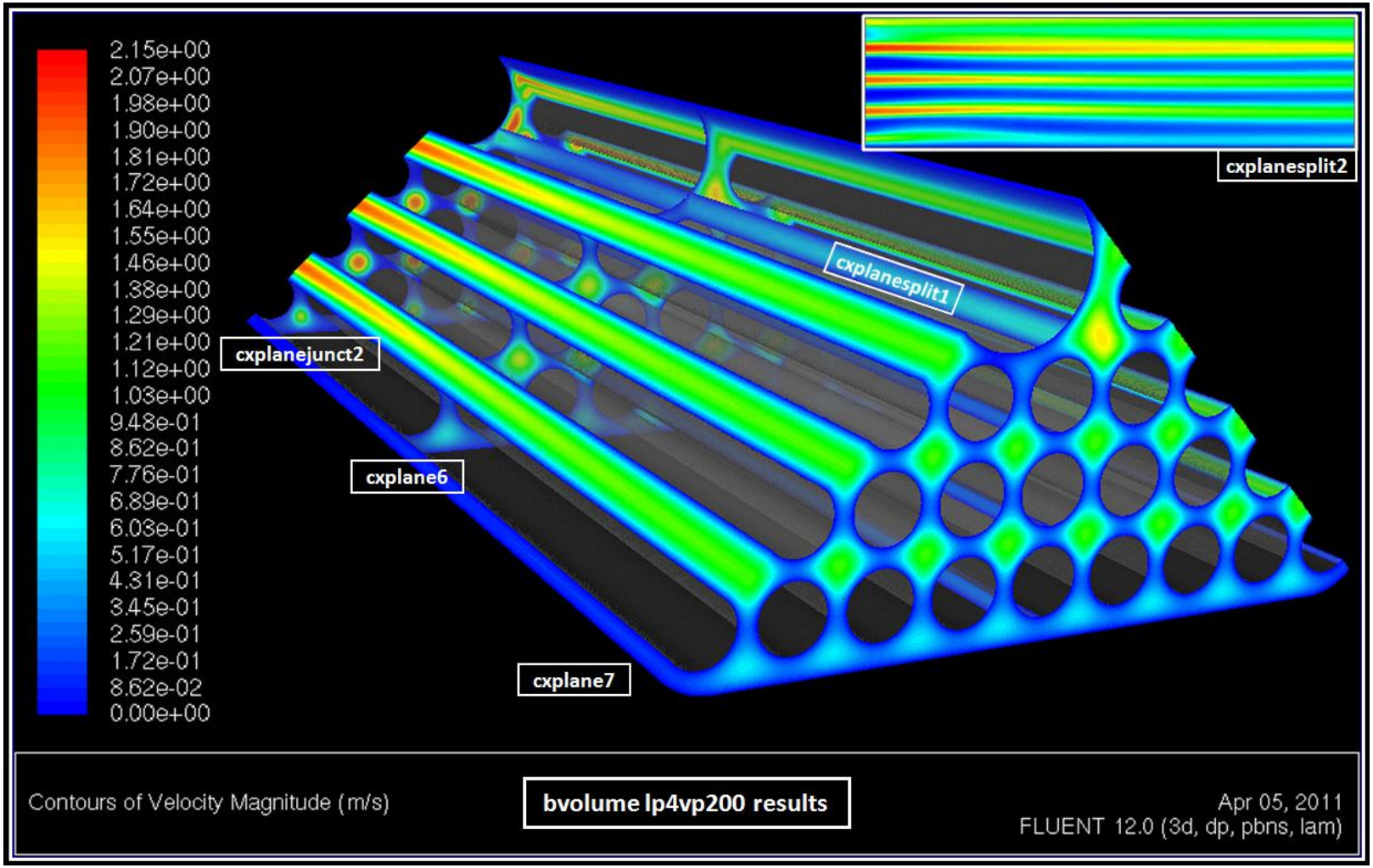


Figure 5.91: Contours of velocity magnitude over specified cross-sections of bvolume for the loop 4 velocity trial at 200 slpm.

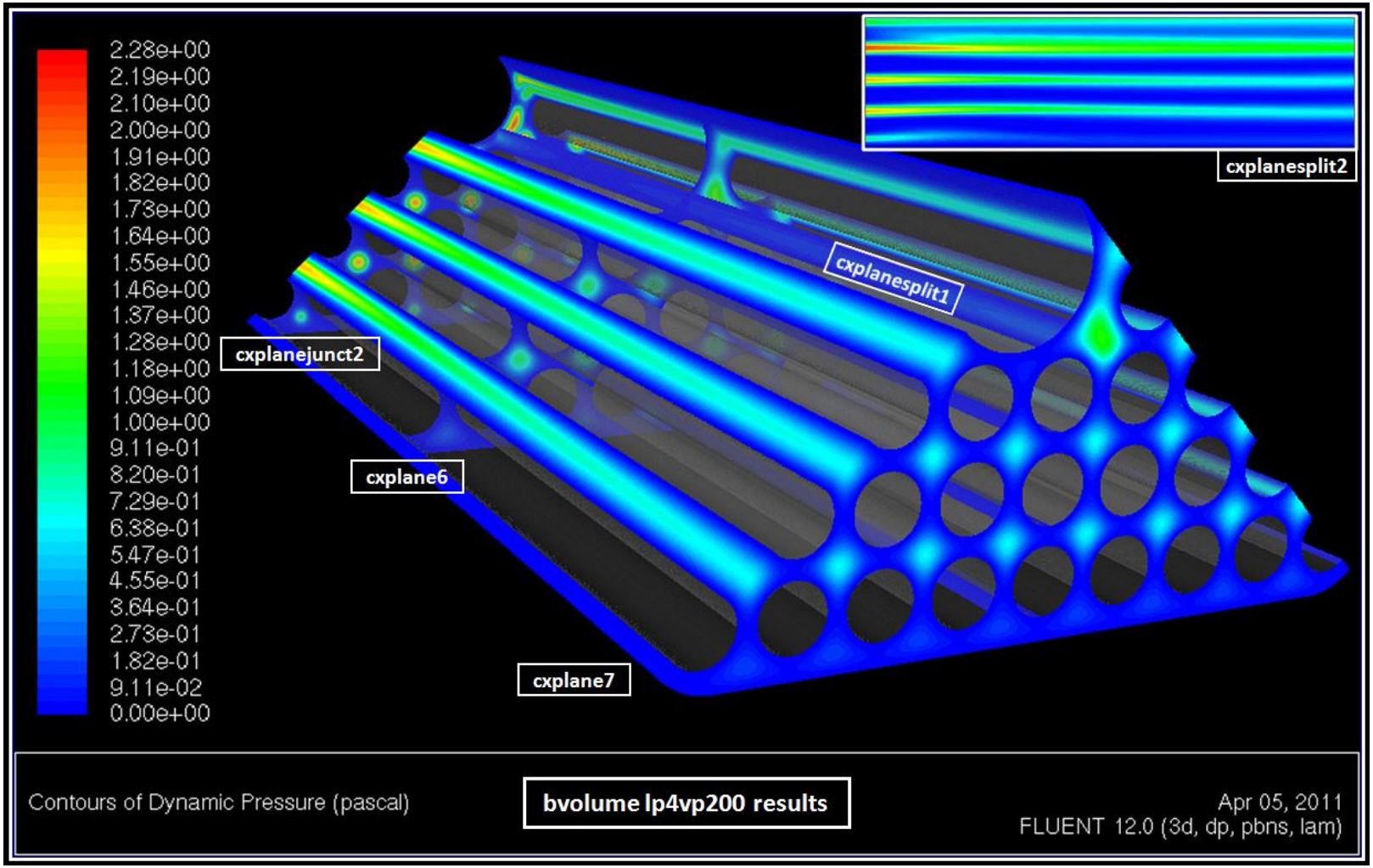


Figure 5.92: Contours of dynamic pressure over specified cross-sections of bvolume for the loop 4 velocity trial at 200 slpm.

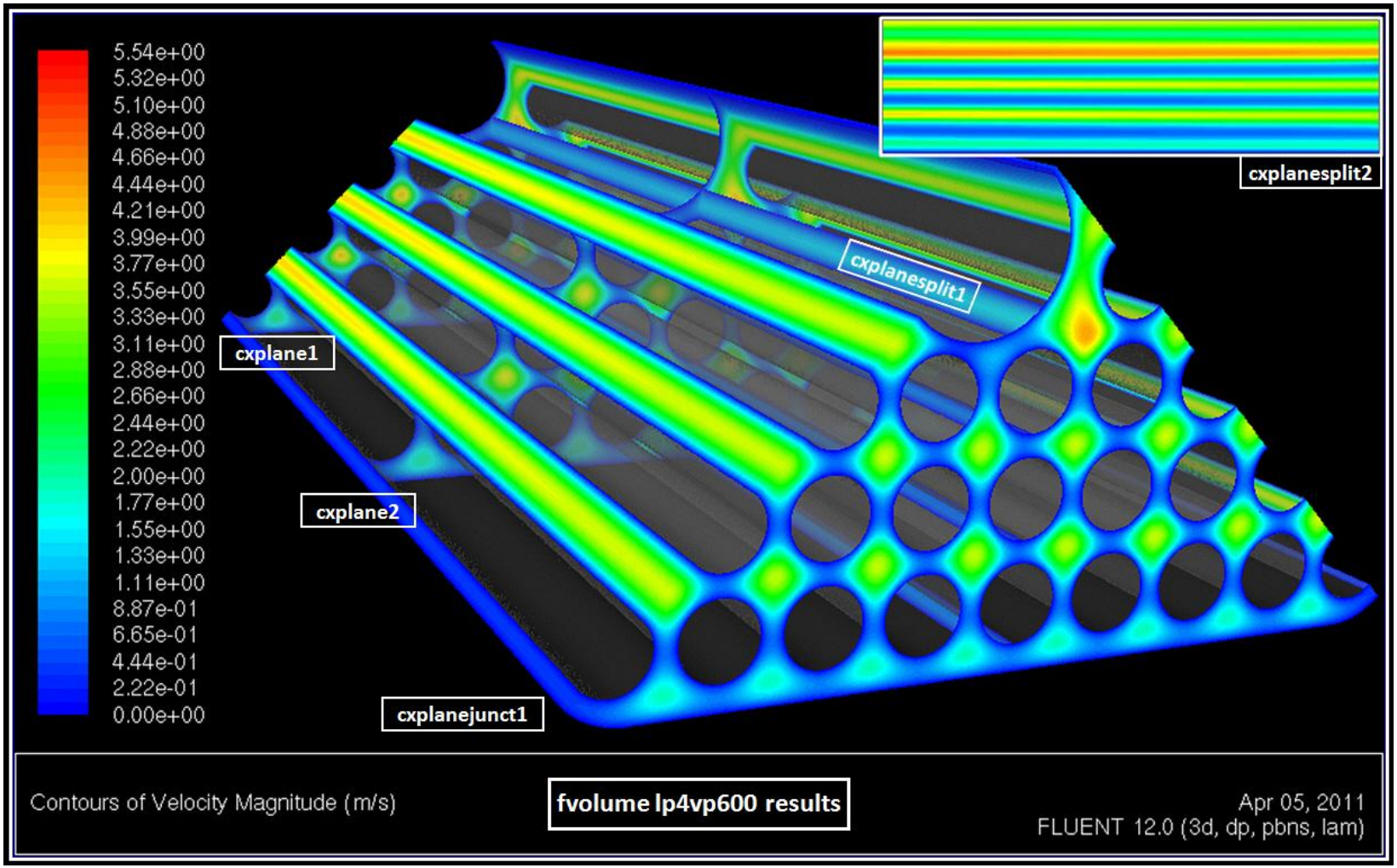


Figure 5.93: Contours of velocity magnitude over specified cross-sections of fvolume for the loop 4 velocity trial at 600 slpm.

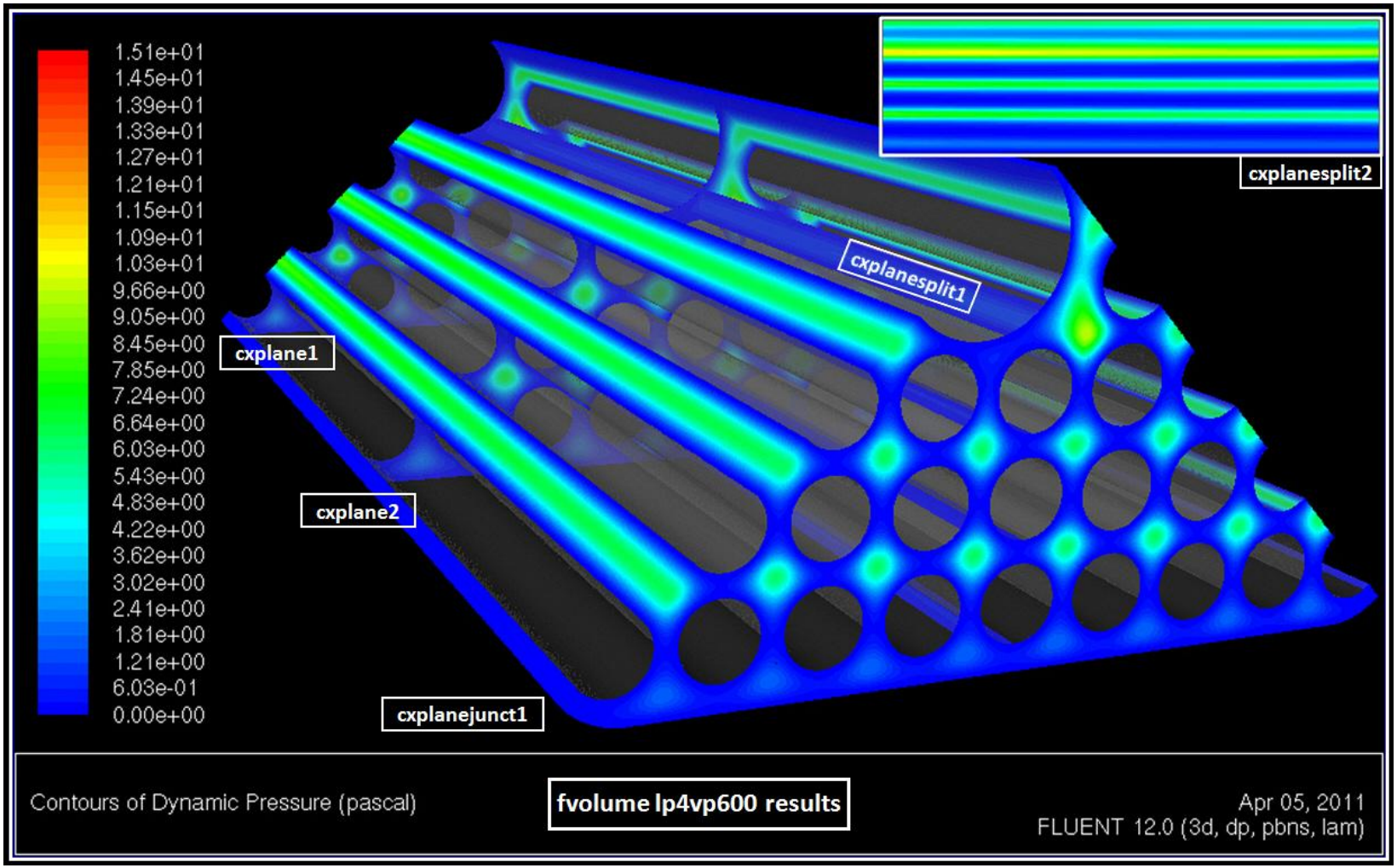


Figure 5.94: Contours of dynamic pressure over specified cross-sections of fvolume for the loop 4 velocity trial at 600 slpm.

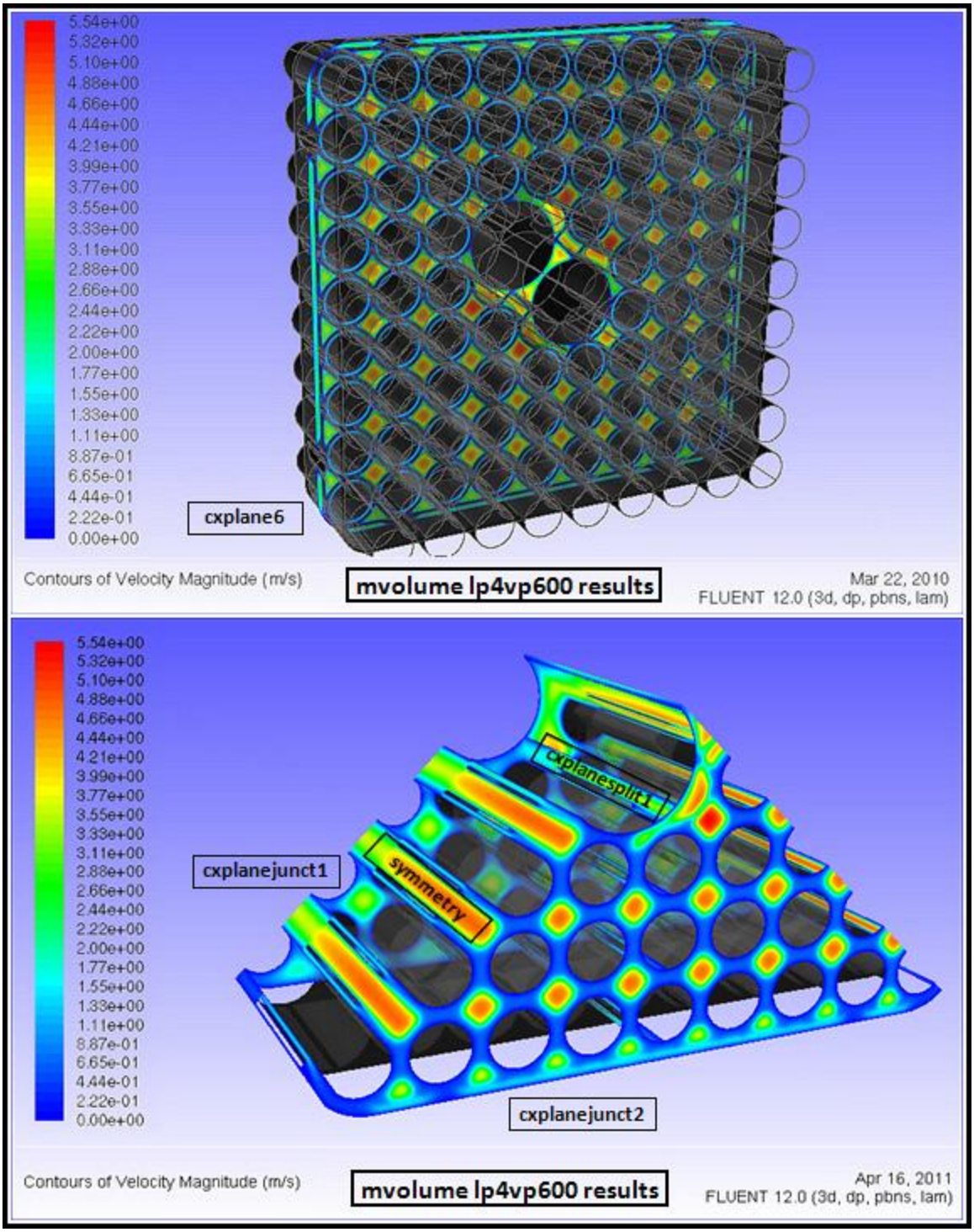


Figure 5.95: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm with and without symmetry.

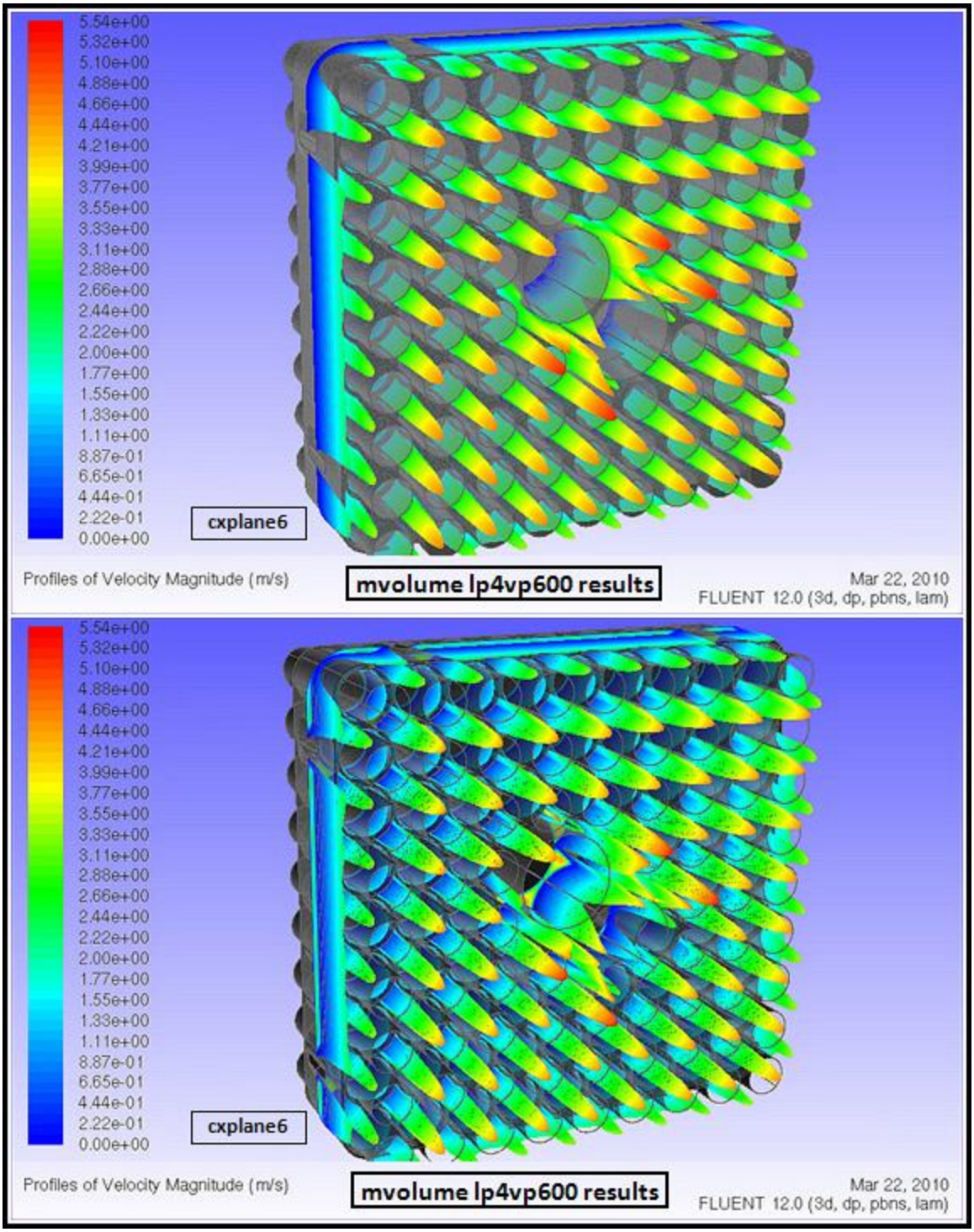


Figure 5.96: Profiles of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm without symmetry.

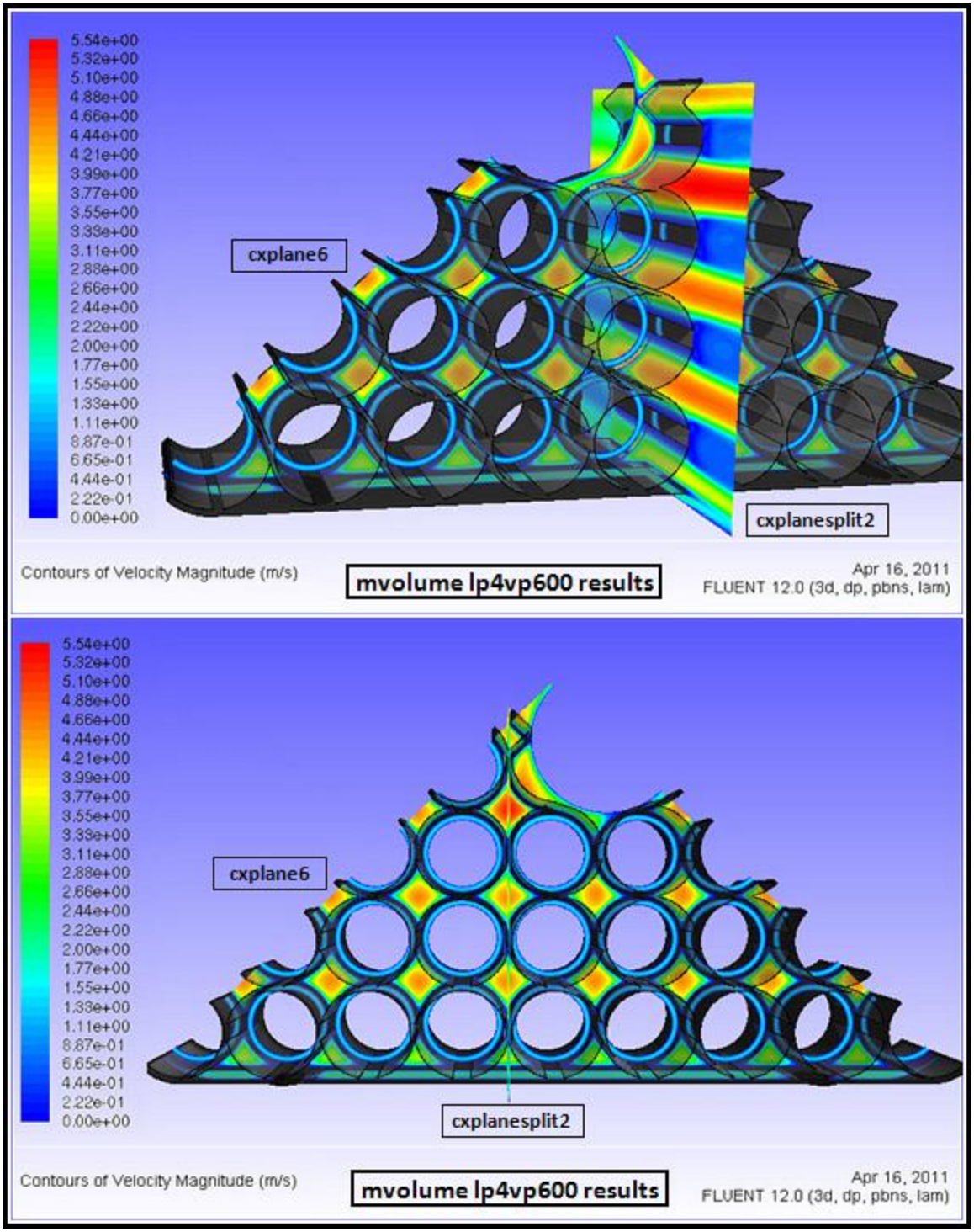


Figure 5.97: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm (1 of 3).

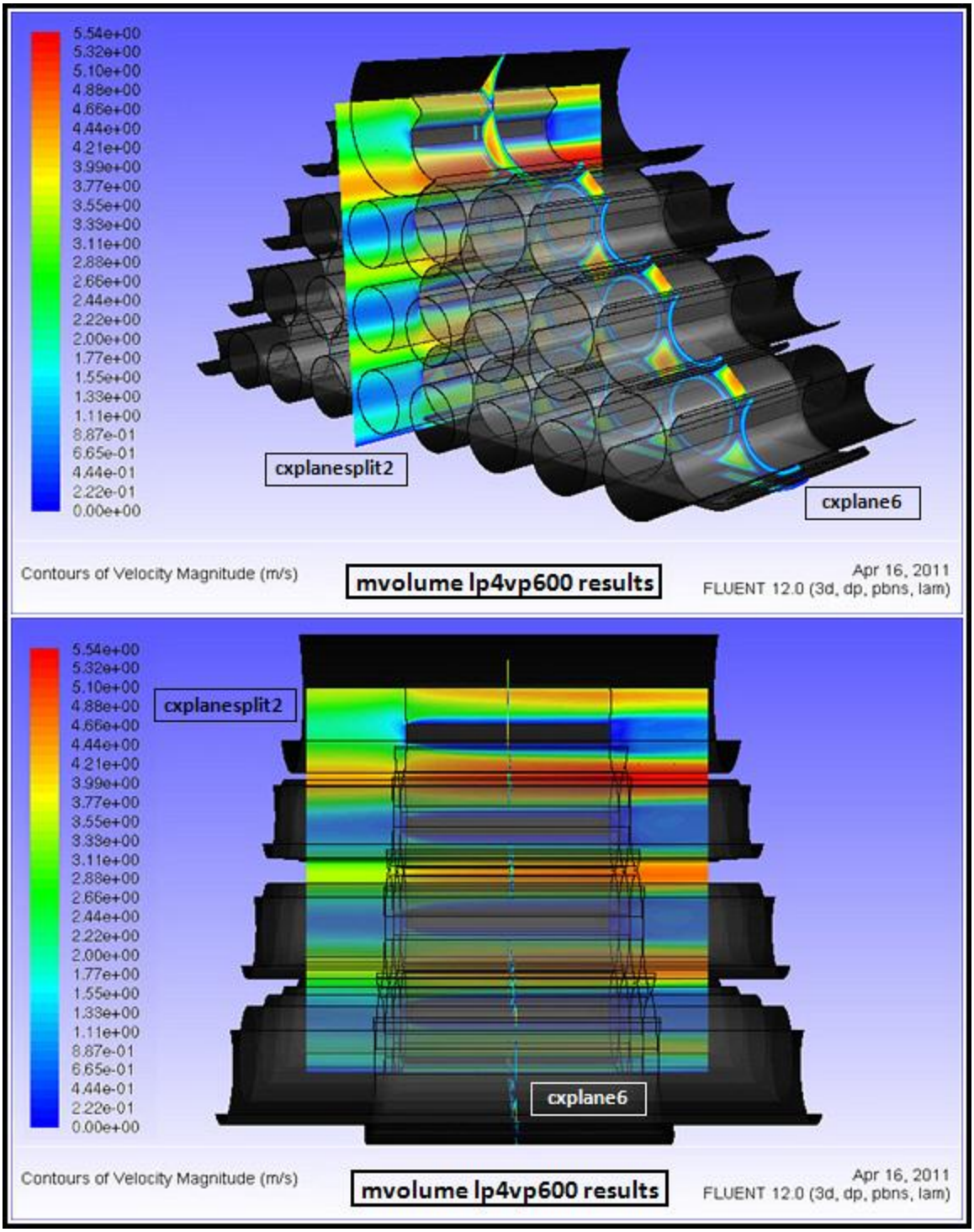


Figure 5.98: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm (2 of 3).

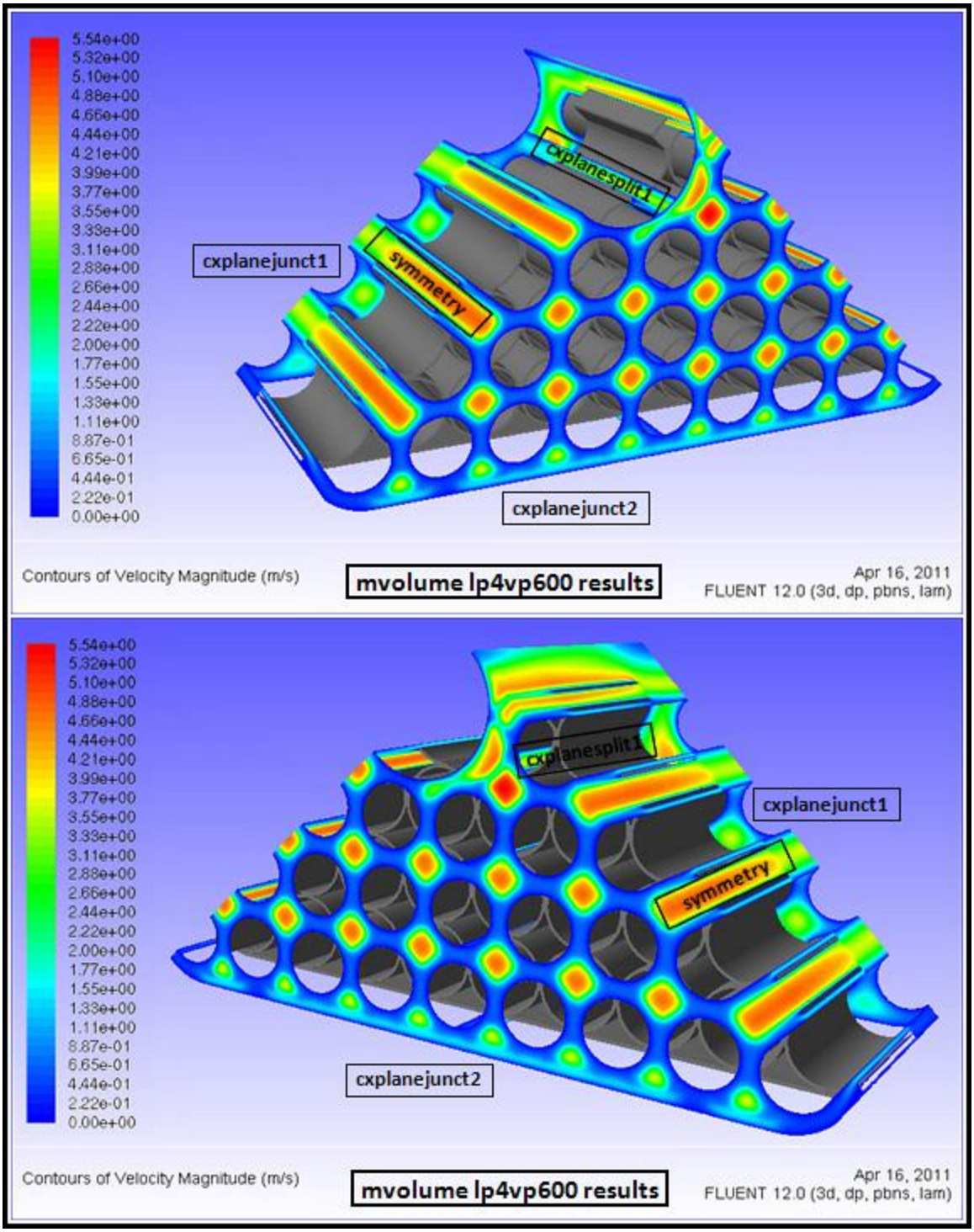


Figure 5.99: Contours of velocity magnitude over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm (3 of 3).

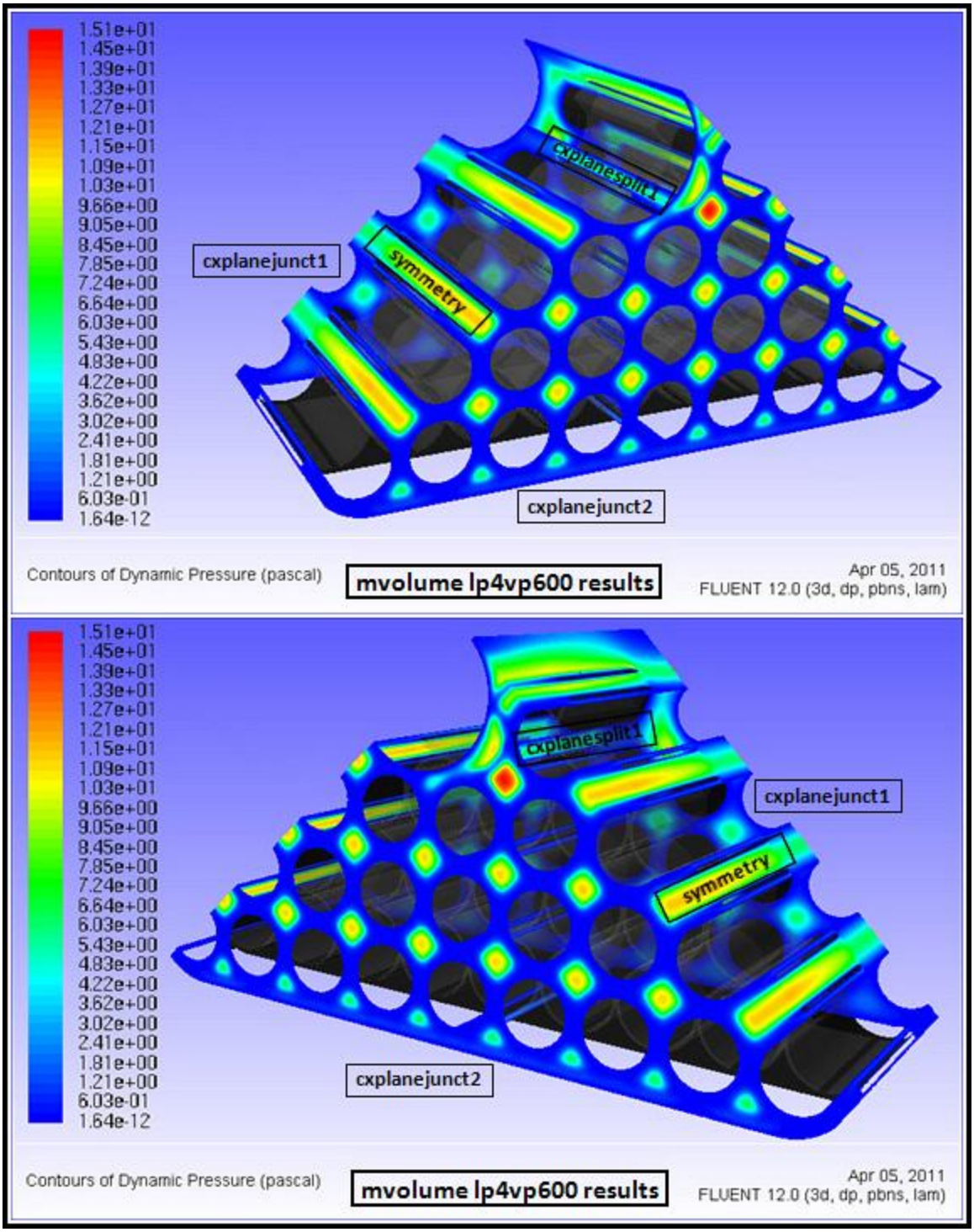


Figure 5.100:Contours of dynamic pressure over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm (1 of 2).

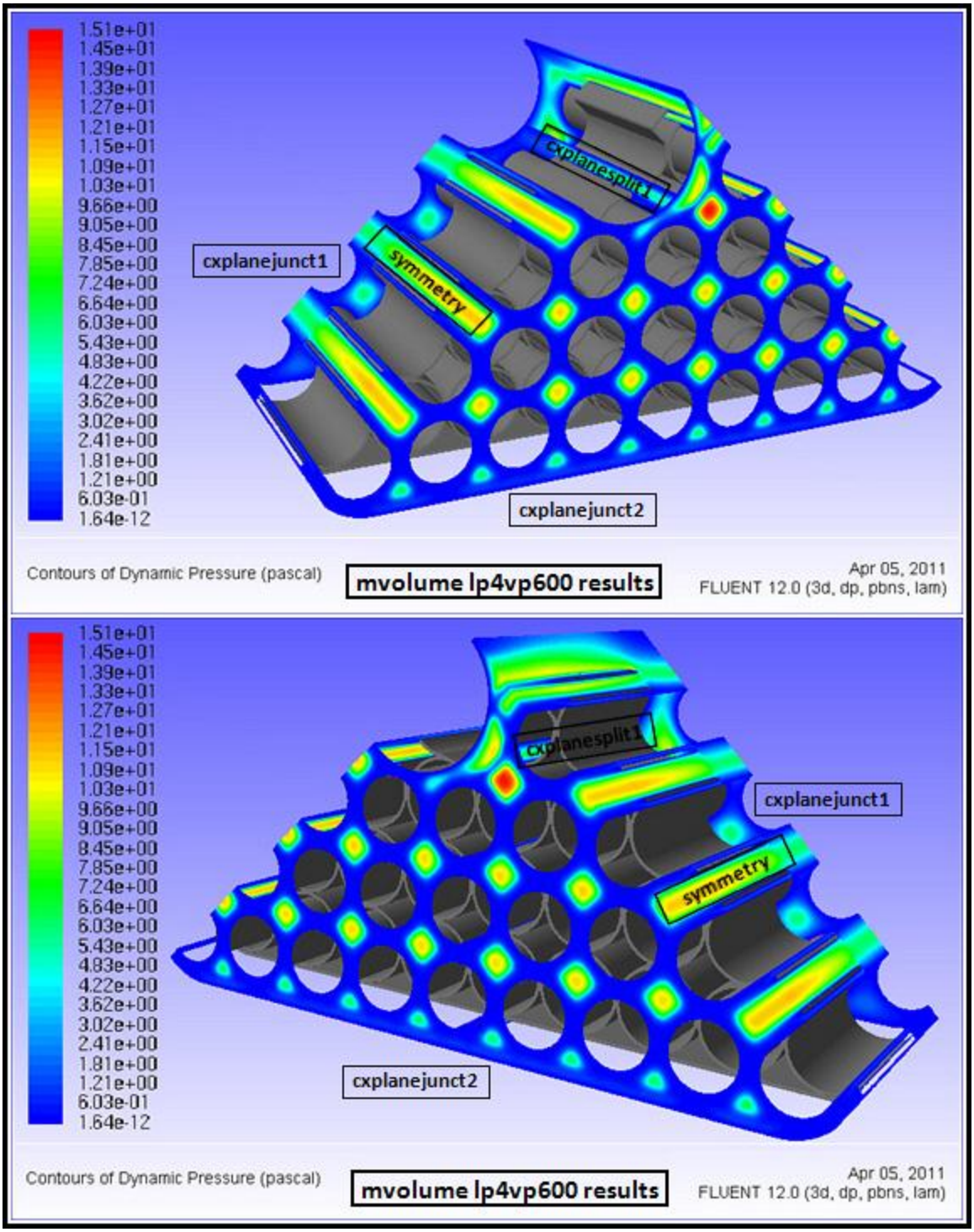


Figure 5.101:Contours of dynamic pressure over specified cross-sections of mvolume for the loop 4 velocity trial at 600 slpm (2 of 2).

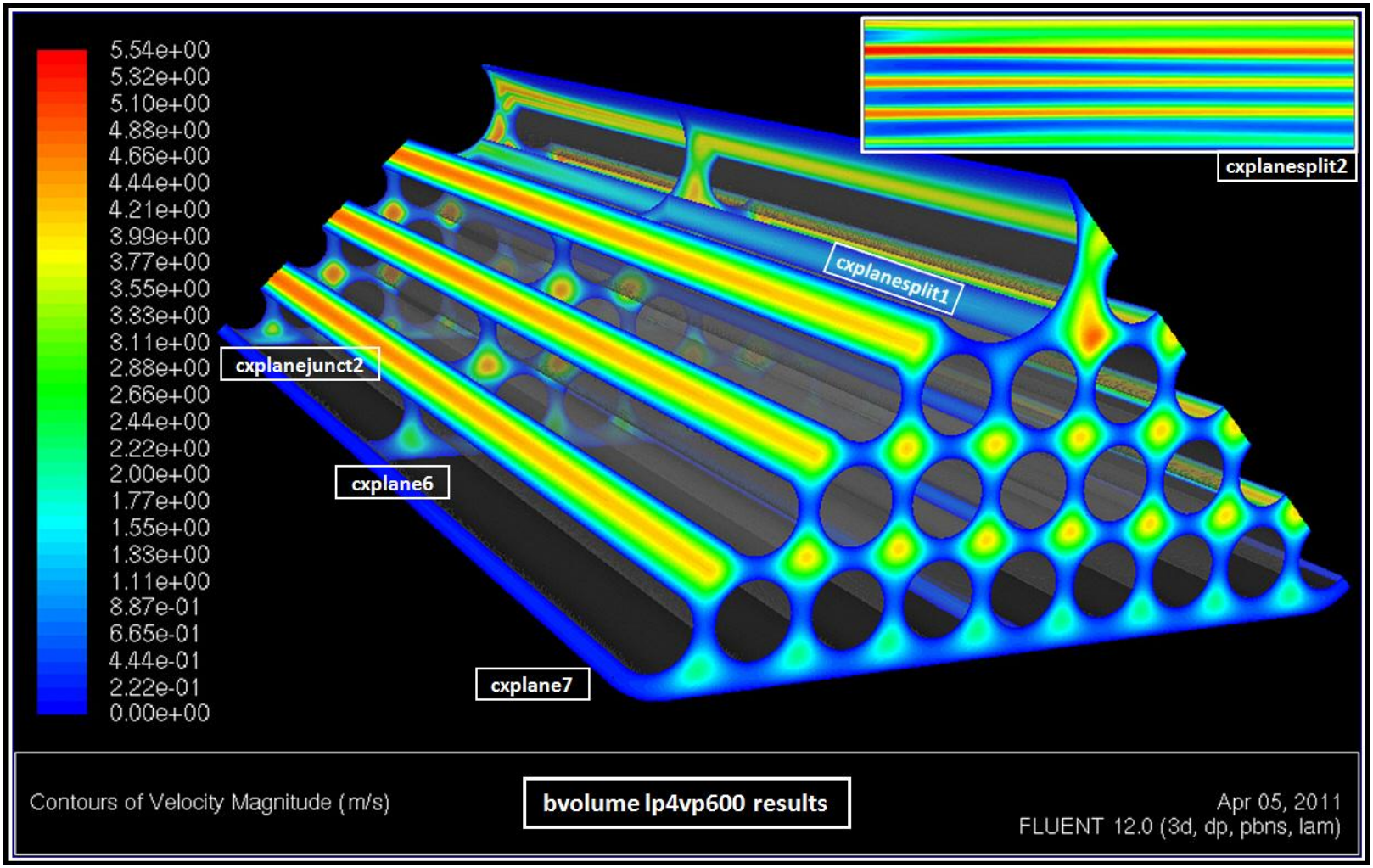


Figure 5.102:Contours of velocity magnitude over specified cross-sections of bvolume for the loop 4 velocity trial at 600 slpm.

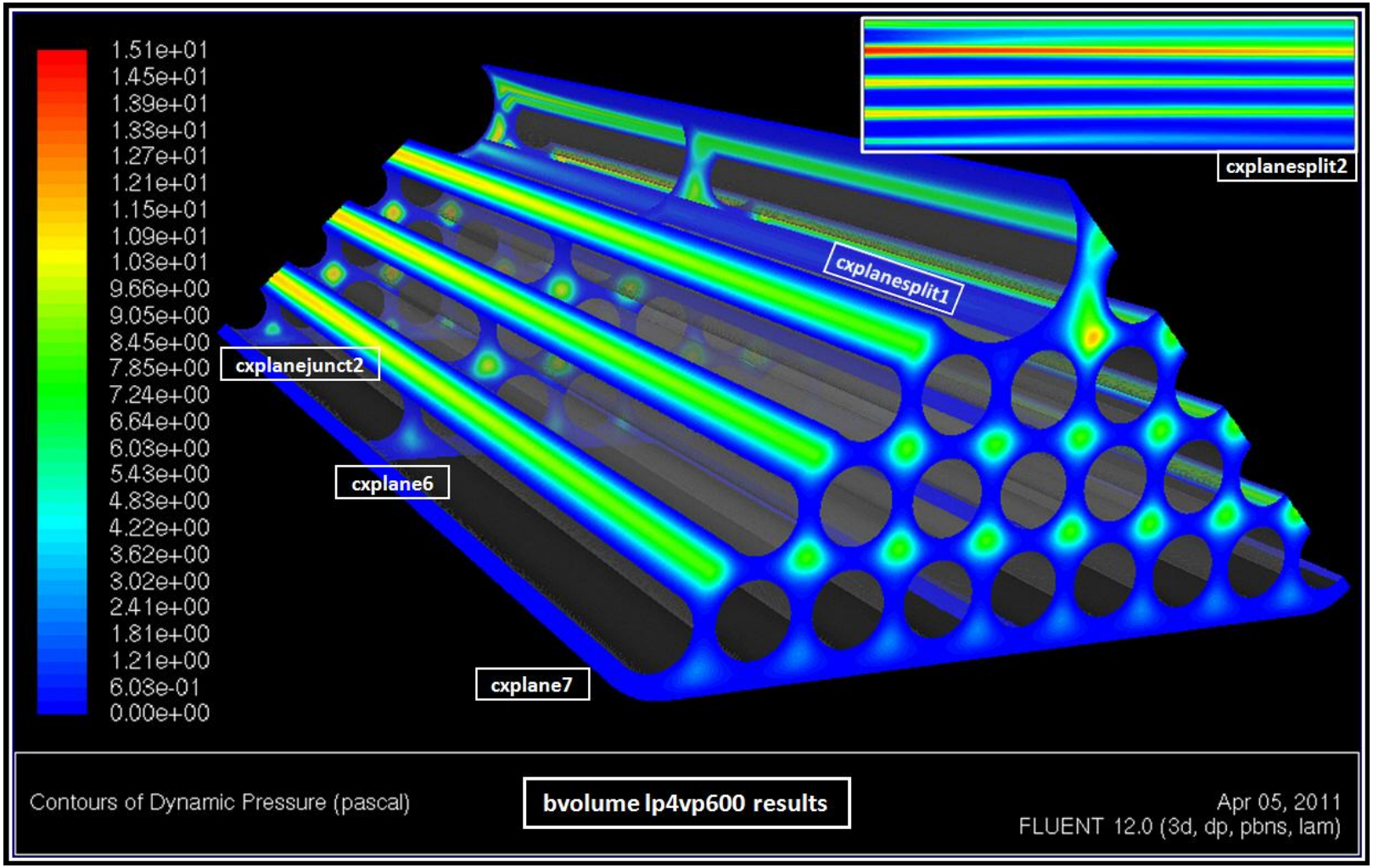


Figure 5.103:Contours of dynamic pressure over specified cross-sections of bvolume for the loop 4 velocity trial at 600 slpm.

5.2.3 Final Comparison of Results

The final comparison of results is directed on conducting a best-case comparative analysis between solutions obtained from the final loop 4 velocity simulations of the fully-populated fuel bundle and spacer model and the SNL experimentally established hydraulic analysis results reported by Durbin and Lindgren (2005). As previously discussed, results from the loop 4 velocity simulations are based on mass-weighted average (MWA) total pressure measurements over specific cross-sectional planes located along the axial length of the modeled domain. An aforementioned number of slight variances in characteristic geometry as well as ambient air properties between the computational simulation and SNL physical experiments (outlined in Chapter 4 and preceding sections of Chapter 5) warrant a certain type of evaluation.

Due to these intentionally implemented as well as unintentionally engendered differences, the airflow velocities for a specified flow rate are not equivalent in both cases, which imply the potential for an incongruent comparison between computational and physical experiment results of pressure loss at the examined flow rates. Therefore, a more suitable comparison is afforded by assessing the hydraulic loss coefficient values from each experimental analysis, which are based on a culmination of measurement solutions over a broad range of flow rates versus a contrast between the individual pressure losses at specific airflow designations. These hydraulic loss coefficients are the primary focus of the following direct comparison with SNL experimental results. The curve fitting of pressure loss data for coefficient values is achieved by employing the TableCurve 2D® automated curve-fitting analysis software (Systat Software Inc., 2011), which fits data to the same equation form used in the SNL hydraulic analysis report by Durbin and Lindgren (2005) as presented in Chapter 3 above.

In order to accomplish a reasonably thorough comparative analysis with SNL report data, the hydraulic loss coefficients from simulation trials are determined for two different segments of the modeled fuel bundle run and spacer domain, which include a spacer only single-component segment as well as a fuel bundle run and spacer multiple-component segment (*i.e.* entire model from inlet to outlet). Similar to the hydraulic analysis performed by SNL, the hydraulic diameter and flow area of a fully-populated spacer are implemented in the curve-fitting analysis for the single-component segment, while the hydraulic characteristics of a fully-populated fuel bundle are incorporated into the multiple-component segment analysis. For the single component involving only the spacer, hydraulic loss coefficients are calculated using two separate segment lengths of which each is defined by the total length between two pressure measurement locations of equivalent upstream and downstream distance from the spacer entity. The first spacer segment length, termed the spacer-junction length, corresponds to the set of pressure measurement planes located 0.001 cm upstream/downstream of the spacer and provides the most accurate representation of pressure losses as well as hydraulic loss coefficient values concerning the fully-populated spacer component.

Although the ideal placement of these pressure measurement planes for this particular spacer-junction segment coincide with the exact interior interfaces at both spacer and fuel bundle junctions, construction of an interior face zone at these locations within the mesh proved implausible due to the compilation method required for assembling the final mesh from multiple constituent part files. As a result, selected locations for the pressure measurement planes at upstream and downstream distances of 0.001 cm from the spacer component ensure that each postprocessing plane created is bounded by cross-sectional extents pertaining to the same area of flow and falls within reasonable proximity to the ideal interface placement. These solution values are

predominantly intended for the evaluation of subsequent research efforts disclosed in Chapter 6 and not the currently discussed assessment with SNL hydraulic analysis results.

The second spacer segment length, labeled the spacer-port length, is attributed to the set of pressure measurement planes located 0.635 cm (or 0.25 inches) upstream and downstream of the fully-populated spacer. This spacer-port length is indicative of the approximate average distance between each pressure port location (immediately adjacent to a spacer) defined in the SNL experimental apparatus (see Figure 3.1) and the nearest upstream or downstream spacer component. Accordingly, the simulation (S_{LAM}) and (k) coefficients derived from pressure losses based on this spacer-port segment length are utilized for comparison with the SNL experimental analysis. Since the only single-component segment of a fully-populated spacer with loss coefficients stated in the SNL hydraulic analysis report pertains to the (13-14) segment in Durbin and Lindgren (2005), the (S_{LAM}) and (k) coefficient results for this segment (shown in Table 3.6 above) are selected for comparison with the simulation values.

Subsequently, for the fuel bundle run and spacer multiple-component segment (*i.e.* entire model domain), the hydraulic loss coefficients are calculated using a single segment length defined by pressure measurement planes coincident with the inlet and outlet model boundaries, which is termed the model-total length. These results are compared with the (S_{LAM}) and (k) coefficient values determined by Durbin and Lindgren (2005) for segment (8-15) as presented in Table 3.5 since the SNL hydraulic analysis report does not include any multiple-component segments that encompass only a fully-populated bundle run and spacer. Although the (8-15) segment constitutes a total of four equivalent fuel bundle run and spacer subsegments, the successively connected and periodically repeating geometry attributed to these subsegments permit the unique

consideration of segment (8-15) loss coefficients as representing an average coefficient set for a bundle run and spacer subsegment that is properly weighted over the entire (8-15) segment length. Hence, the actual (S_{LAM}) and (k) coefficients compared with the computational simulations are based upon segment (8-15) values in Table 3.5 from the SNL hydraulic analysis report, but appropriately weighted on a per bundle run and spacer subsegment basis. Specifically, the (S_{LAM}) coefficient remains unchanged from segment (8-15) since the value is already weighted on a per unit length corresponding to a fuel bundle run and spacer subsegment, while the (k) coefficient from segment (8-15) necessitates weighting based on a per unit number of total bundle run and spacer subsegments encompassed by the (8-15) segment path (*i.e.* four subsegments).

Finally, despite the slightly inexact comparison of results for a stated airflow rate, the total pressure losses across the fully-populated fuel bundle and spacer model from loop 4 velocity simulation runs are examined in conjunction with the approximated pressure loss measurements from SNL experimental trials. The SNL pressure losses are estimated for an equivalent fuel bundle run and spacer segment using the weighted coefficient values determined above from the (8-15) segment results as well as reported hydraulic flow characteristics (see Table 3.5) provided by Durbin and Lindgren (2005). Results of the aforementioned final comparison between loop 4 velocity simulations and the SNL hydraulic analysis report are numerically tabulated in Tables 5.10 through 5.13 below, which include the general reference values as well as equations fundamental to the calculated solutions. Furthermore, pressure losses across the entire fuel bundle run and spacer segment length from both computational and SNL experimentally based measurements are plotted against the respective fuel bundle velocities in Figure 5.104 as an additional graphical comparison of the results.

Table 5.10: Final comparison of results between loop 4 velocity simulations and the SNL hydraulic analysis report by Durbin and Lindgren (2005) including general reference values as well as equations used in the calculation of loss coefficient solutions (1 of 4).

FINAL COMPARISON OF LOOP 4 VELOCITY SIMULATIONS AND SNL HYDRAULIC ANALYSIS RESULTS INCLUDING REFERENCE VALUES AND CALCULATIONS

[General Reference Values And Curve-Fit Equations For Hydraulic Loss Coefficients: Fully-Populated Bundle Run And Spacer Model]

Air Properties	Value:	Fuel Bundle $_{x\text{-section}}$	Value:	Spacer $_{midplane\ x\text{-section}}$	Value:	Segment Lengths - L	Value:
Density (kg/m^3)	0.98141470	Flow Area (m^2)	0.00968829	Flow Area (m^2)	0.00726749	Spacer-Junction (m)	0.030500
Dynamic Viscosity (kg/m-s)	1.820975E-05	Hydraulic Diameter (m)	0.01190721	Hydraulic Diameter (m)	0.00290797	Spacer-Port (m)	0.043180
						Model-Total (m)	0.511810

Note: The segment length value represents the distance between the two cross-sectional planes where pressure measurements are taken to calculate a segment pressure loss, which is used in the curve-fit equations as (L); it does not necessarily equal the actual length of the segment solid component(s) in the model. Two segment lengths are used for the spacer loss calculations. The spacer-junction length is a more accurate representation of losses across the spacer since pressure measurements are taken at planes located 0.001 cm up/downstream of the component; these results are important for discussions in Chapter 6. However, the spacer-port length represents an approximate average distance for measurement port locations up/downstream of the spacers (~ 0.635 cm or 0.25 inches) designated in the SNL experiment; these results are appropriately used for comparisons with SNL hydraulic analysis data as discussed in the current section of Chapter 5.

Pressure Loss Equation Fitted

$$\Delta P_{total} = a_2 V^2 + a_1 V$$

Linear And Quadratic Fit Terms

$$a_1 = \frac{S_{LAM} \cdot L \cdot \mu}{2 \cdot D_H^{\,2}} \qquad a_2 = \frac{\rho \cdot \sum k}{2}$$

Viscous And Form Hydraulic Loss Coefficients

$$S_{LAM} = 2a_1 \left(\frac{D_H^2}{L\mu} \right) \qquad \sum k = \frac{2a_2}{\rho}$$

Note: The curve-fit equations are based on the SNL hydraulic analysis report by Durbin and Lindgren (2005); see the explanations/descriptions provided in Chapter 3. In addition, all simulation pressure losses are based on mass-weighted-average (MWA) total pressure measurements at designated cross-sectional planes as previously emphasized. For segments involving both a fuel bundle run and spacer, the curve-fit data and loss coefficient calculations are based on the fuel bundle hydraulic diameter and flow area as in the SNL hydraulic analysis report.

Table 5.11: Continuation of the final comparison of results between loop 4 velocity simulations and the SNL hydraulic analysis report by Durbin and Lindgren (2005) including general reference values as well as equations used in the calculation of loss coefficient solutions (2 of 4).

[Hydraulic Loss Coefficient Results: Spacer Segment Defined By Spacer-Junction Length]

Simulation Curve-Fit Data

Flow Rate (slpm)	200	300	400	500	600
Spacer Velocity (m/s)	0.60503955	0.90755932	1.21007910	1.51259887	1.81511864
Pressure Loss (Pa)	0.84448180	1.37478210	1.95124840	2.56059080	3.19563240

TableCurve 2D Curve-Fit And Loss Coefficient Results

a_1 (N-s/m^3):	a_2 (N-s^2/m^4):	R^2 ():	S_{LAM} ():	$\sum k$ ():
1.264430	0.277359	0.999691	**38.503605**	**0.565222**

Note: Again, the above results are primarily important to discussions in Chapter 6; the results considered for current discussions of comparison to SNL hydraulic analysis data follow below.

[Hydraulic Loss Coefficient Results: Spacer Segment Defined By Spacer-Port Length]

Simulation Curve-Fit Data

Flow Rate (slpm)	200	300	400	500	600
Spacer Velocity (m/s)	0.60503955	0.90755932	1.21007910	1.51259887	1.81511864
Pressure Loss (Pa)	0.91854470	1.48280470	2.09509610	2.74263960	3.42057420

TableCurve 2D Curve-Fit And Loss Coefficient Results

a_1 (N-s/m^3):	a_2 (N-s^2/m^4):	R^2 ():	S_{LAM} ():	$\sum k$ ():
1.379433	0.281737	0.999793	**29.670467**	**0.574145**

Comparison To SNL Segment 13-14 Results From Durbin and Lindgren (2005); See Table 3.6

Loss Coefficient	FLUENT-Simulations	SNL-Analysis	%-Difference
S_{LAM} ()	29.670467	32	**7.2798**
$\sum k$ ()	0.574145	1.8	**68.1030**

Table 5.12: Continuation of the final comparison of results between loop 4 velocity simulations and the SNL hydraulic analysis report by Durbin and Lindgren (2005) including general reference values as well as equations used in the calculation of loss coefficient solutions (3 of 4).

[Hydraulic Loss Coefficient Results: Fuel Bundle Run And Spacer Segment Defined By Model-Total Length]

Simulation Curve-Fit Data

Flow Rate (slpm)	200	300	400	500	600
Bundle Velocity (m/s)	0.45385897	0.68078845	0.90771794	1.13464742	1.36157691
Pressure Loss (Pa)	2.31055800	3.60407481	4.95473410	6.34855600	7.77216090

TableCurve 2D Curve-Fit And Loss Coefficient Results

a_1 (N-s/m^3):	a_2 (N-s^2/m^4):	R^2 ():	S_{LAM} ():	$\sum k$ ():
4.870801	0.624417	0.999903	**148.196426**	**1.272484**

Comparison To SNL Segment 8-15 Based Results From Durbin and Lindgren (2005); See Table 3.5

Loss Coefficient	FLUENT-Simulations	SNL-Analysis	%-Difference
S_{LAM} ()	148.196426	137	**8.1726**
$\sum k$ ()	1.272484	3.75	**66.0671**

Note: Although the SNL loss coefficients for segment 8-15 in Table 3.5 represent a total length comprised of 4 fully-populated fuel bundle and spacer subsegments, their successively connected and periodically repeating geometry permit these two loss coefficient values to also be appropriately considered as a representative average of equivalent coefficient sets from each of the 4 individual fuel bundle and spacer subsegments, which are properly weighted over the entire 8-15 segment length to provide the actual loss coefficient values shown in Table 3.5. Thus, for a single equivalent fuel bundle and spacer subsegment, the viscous hydraulic loss coefficient is the same as the value given in Table 3.5 for the total 8-15 segment associated with 4 fuel bundle and spacer subsegments since the viscous hydraulic loss coefficient value is already weighted on a per fuel bundle and spacer unit length as shown above in the corresponding coefficient equation. On the other hand, the form hydraulic loss coefficient is given as a sum for the total number of fuel bundle and spacer subsegments attributed over segment 8-15 as shown in its corresponding coefficient equation above. Therefore, the value given in Table 3.5 must be divided by 4 to obtain the correct form loss coefficient value weighted per single fuel bundle and spacer unit. This outlines how the SNL values used in the comparison above are obtained for a single fuel bundle and spacer constituent using the total segment 8-15 results shown in Table 3.5 from the SNL hydraulic analysis report.

Table 5.13: Continuation of the final comparison of results between loop 4 velocity simulations and the SNL hydraulic analysis report by Durbin and Lindgren (2005) including general reference values as well as equations used in the calculation of loss coefficient solutions (4 of 4).

[SNL Hydraulic Analysis Pressure Loss Comparison: Fuel Bundle Run And Spacer Segment]

SNL Hydraulic Loss Coefficient Data For Segment 8-15 From Durbin And Lindgren (2005); See Table 3.5

Density (kg/m^3):	Viscosity (kg/m-s):	Segment Length (m):	Flow Area (m^2):	Hydraulic Diameter (m):	S_{LAM} ():	∑k ():
0.98	1.85E-05	2.05	0.0098	0.0119	137	15

SNL Hydraulic Loss Coefficient Data - Properly Weighted Per Fuel Bundle Run And Spacer Unit Segment

Density (kg/m^3):	Viscosity (kg/m-s):	Segment Length (m):	Flow Area (m^2):	Hydraulic Diameter (m):	S_{LAM} ():	∑k ():
0.98	1.85E-05	0.5125	0.0098	0.0119	**137**	**3.75**

Fuel Bundle Run And Spacer Segment Pressure Loss Calculations Based On SNL Hydraulic Loss Coefficient Data

Flow Rate (slpm)	200	300	400	500	600
Bundle Velocity (m/s)	0.44933323	0.67399984	0.89866646	1.12333307	1.34799969
Pressure Loss (Pa)	2.43176788	3.92589573	5.60551954	7.47063929	9.52125499

Note: The pressure losses are calculated using the curve-fit equations shown above, which are based on the SNL hydraulic analysis report by Durbin and Lindgren (2005) discussed in Chapter 3.

Comparison Between SNL And FLUENT Simulation Pressure Losses Over A Fully-Populated Fuel Bundle Run And Spacer Segment

Flow Rate (slpm)	200	300	400	500	600
FLUENT Bundle Velocity (m/s)	0.45385897	0.68078845	0.90771794	1.13464742	1.36157691
SNL Bundle Velocity (m/s)	0.44933323	0.67399984	0.89866646	1.12333307	1.34799969
FLUENT Pressure Loss (Pa)	2.31055800	3.60407481	4.95473410	6.34855600	7.77216090
SNL Pressure Loss (Pa)	2.43176788	3.92589573	5.60551954	7.47063929	9.52125499
Difference (%)	**4.9844**	**8.1974**	**11.6097**	**15.0199**	**18.3704**

Note: These pressure losses are plotted with respect to the corresponding fuel bundle velocities in the subsequent graphical figure for further comparison.

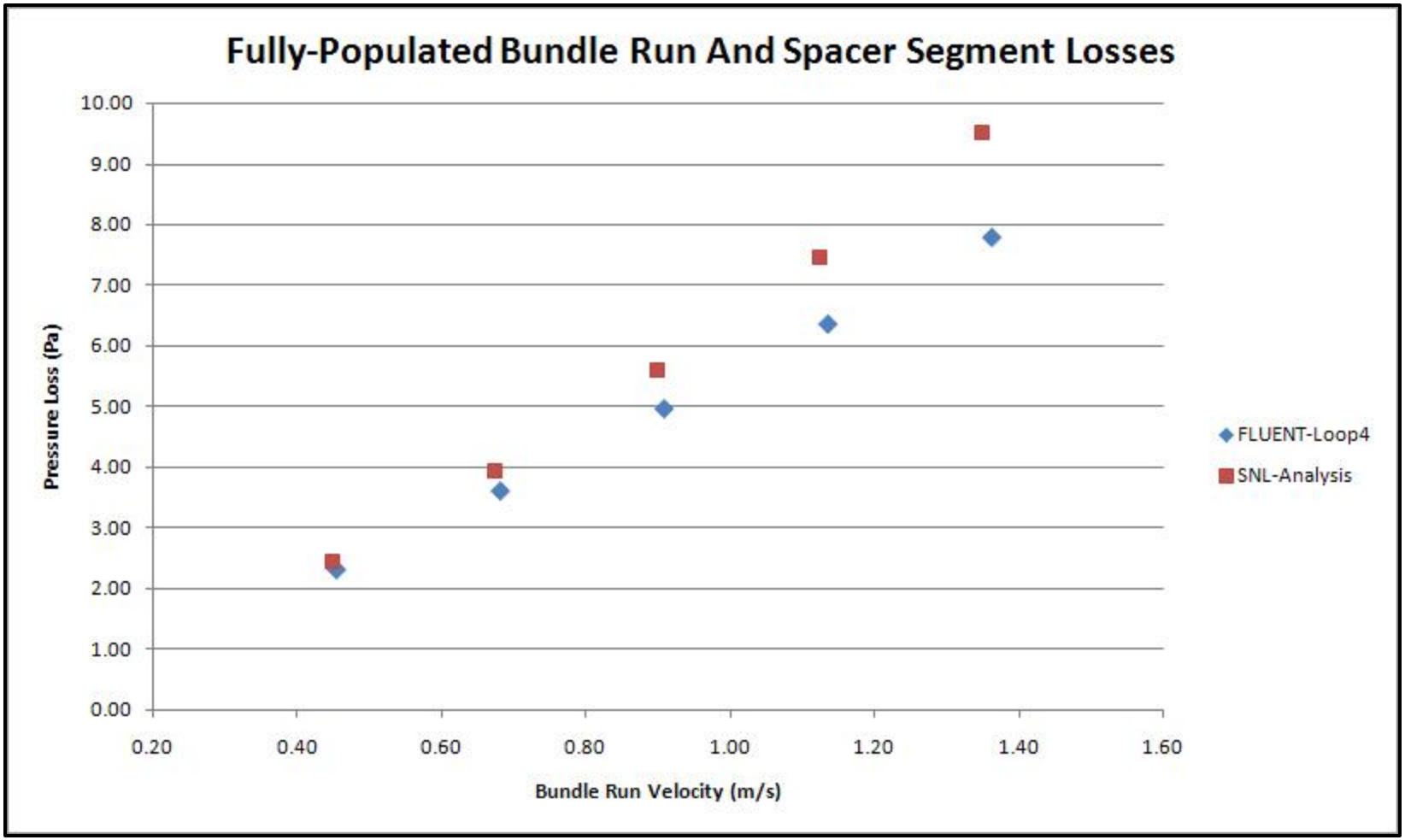


Figure 5.104:Comparison of pressure losses across entire fuel bundle run and spacer segment from loop 4 simulations and approximations to SNL experimental measurements based on the coefficient data in Durbin and Lindgren (2005).

The final comparison of results presented in Tables 5.10 through 5.13 above clearly illustrates several discrepancies between the hydraulic loss coefficient values obtained from loop 4 velocity simulation trials and the SNL hydraulic analysis report by Durbin and Lindgren (2005) for both a single-component spacer as well as multiple-component fuel bundle and spacer segment of the fuel assembly domain. As previously mentioned, the slight variances in characteristic assembly dimensions, ambient airflow properties, and pressure measurement details concerning computational and SNL physical experiments account for an extent of the inconsistencies observed. However, even with additional consideration for uncertainties in the SNL experimental measurements, which are reported by Durbin and Lindgren (2005) as ($S_{LAM} \pm 5$) and ($k \pm 1$), a reasonably conservative estimate of the maximum percent difference between computational and SNL loss coefficients (attributed to these variances) is no more than 10% to 15% of the values compared.

In other words, percent differences for loss coefficient comparisons that exceed this maximum estimate are likely the result of error(s) incurred beyond the scope of acknowledged experimental variances. Since this condition exists for several loss coefficients compared in Tables 5.10 through 5.13 above, ascribing responsibility for the large deviations in respective coefficient values to other potential sources of error outside these slight variances among the computational and SNL physical experiments is certainly warranted. Therefore, an attempt to provide probable explanation(s) for the above solution discrepancies with a more in-depth analysis of the results is subsequently addressed in discussions that follow.

In order to accomplish this objective in the most meaningful and straightforward manner, an approach is adopted for presenting further considerations of the compared results whereby pertinent observations are distinctly outlined based on the assumptions

stemming from one of two unique perspectives. The first perspective, labeled perspective #1, assumes that SNL experimental measurements as well as hydraulic analysis results represent the ideal case values. Thus, observations characterized by perspective #1 are derived under the hypothetical circumstance in which the computational model and simulation trials are the predominant sources of error corresponding to the large differences in specific loss coefficient values. The second perspective, termed perspective #2, entertains the possibility that SNL experimental measurements as well as hydraulic analysis results are not indicative of ideal loss coefficient values. Hence, general observations based on perspective #2 are listed under the assumption that a quantitatively significant degree of error represented by the large discrepancies among certain loss coefficient comparisons is due to additional deficiencies in the SNL experiment and hydraulic analysis of measurement data.

These two perspectives embody multiple considerations from both extremes as well as a number of highly probable explanations for the percent differences that exceed acceptable limits of known experimental uncertainty. Lacking any considerable form of direct involvement in the computational modeling efforts, the observational points derived from perspective #1 are intended to provide the SNL researchers with essential information and factors of noteworthy deliberation with regard to the simulation trials performed. Aptly combined by SNL researchers with recognized issues on their behalf surrounding the physical experiments, these supplemental points of interest impart a more thorough understanding and knowledge of the completed research under current assessments as well as the contemplation of future SNL research initiatives. As a result of the substantial application and benefit afforded to SNL, initial focus is directed upon the notable observations as well as solution explanations that are based on perspective #1 (and related assumptions), which are summarized in a bulleted list format below.

Key Observations of the Results Analysis from Perspective #1:

- The hydraulic loss coefficient values for the viscous (S_{LAM}) terms of both the fully-populated single-component spacer and multiple-component fuel bundle with spacer segments are sufficiently below the maximum acceptable limit of 10% to 15% difference with compared values from the SNL hydraulic analysis. Conversely, the inertial (k) terms for both of these segments are well-outside the acceptable range of percent difference with the reported values for SNL equivalents; thus, explanations for solution discrepancies among these terms are of primary concern.
- Sizeable differences between upstream and downstream distances of pressure ports to an adjacent spacer within the computational model and SNL experimental apparatus are one possible consideration for large variances in the (k) solutions since static pressure gradients shown in Figure 5.80 are relatively steep near the spacer component. However, this graphical feature is a clear indication of potential errors introduced only as a consequence of inconsistent port locations and the neglect of kinetic energy differences within curve-fit derivations (for loss coefficients) employing static pressure measurements near spacer components; it is not directly applicable toward explanations concerning MWA measurements of total pressure as utilized in the computational hydraulic analysis.
- Although SNL experimental measurements of static pressure are susceptible to the aforementioned, basing observations on the current perspective assumes that solutions from the SNL hydraulic analysis report are representative of the ideal case values. Hence, deliberations with regard to this subject are addressed under assumptions allied to the observational analysis from the second perspective.

- Nonetheless, selected pressure port locations within the vicinity of a spacer component still affect loss coefficient values calculated for the encompassed segment even though the error effected using MWA total pressures over cross-sectional planes at measurement points is less pronounced since the kinetic energy is properly accounted for in the system energy balance. An indication of the imposed influence on loss coefficient values is demonstrated by comparing the results obtained in Tables 5.10 through 5.13 for two spacer segments with varying pressure port locations upstream and downstream of the spacer component, which are defined by the spacer-junction and spacer-port lengths.
- Contrasting the loss coefficients derived for each spacer segment length reveals a clear differentiation between the respective (S_{LAM}) values, but only a minimal change among the (k) term solutions. Therefore, evidence suggests that a percentage of the difference between compared spacer segment (S_{LAM}) values from the computational and SNL physical experiments is potentially attributable to the variation in pressure port locations, but fails to provide an explanation for the large discrepancy among (k) coefficient results.
- Furthermore, with regard to the computational model only, diverse specifications for the upstream and downstream distances of pressure ports from an adjacent spacer component lack any direct bearing on the loss coefficients determined for the entire fuel bundle and spacer domain segment defined by the placement of current model boundaries. Hence, the substantial difference between computational and SNL hydraulic analysis (k) values for compared fuel bundle and spacer length segments exists regardless of the pressure port locations that internally span the spacer component within the computational domain model.

- A second possible explanation to consider for the large variances in (k) solution values stems from the assumption of periodic flow conditions across model system boundaries. Initially, the accuracy of imposing such boundary conditions for representing the actual flow field established across a fully-populated fuel bundle and spacer segment of the assembly domain is apparently refuted by decreasing similarity between the SNL experimental pressure losses shown in Figure 5.104 and those corresponding to each successive velocity loop trial illustrated in Figure 5.53 above. In fact, the pressure losses from velocity loop 1 are within acceptable limits of experimental uncertainty with reported results for the SNL pressure loss measurements.
- However, for the inlet boundary conditions examined in Tables 5.6 and 5.7 concerning the fuel bundle run model, derived SNL pressure losses contrast sharply with the simulation results based on a uniform inlet velocity profile (closely resembling the fuel bundle and spacer model profile imposed for loop 1). Conversely, derived SNL pressure losses compare favorably (sufficing acceptable limits of percent difference) with the simulation results based on a fuel bundle fully-developed/periodic velocity profile (closely resembling the velocity profile development correlated with loop 4 of the fuel bundle and spacer model).
- Moreover, remarkable agreement between the (S_{LAM}) value in Table 5.11 for the spacer segment of defined spacer-junction length with boundary profiles approaching periodic flow-field conditions and the corresponding (S_{LAM}) value for a near-equivalent spacer segment, which is independently calculated using the approximation method discussed in Section 6.4.3, further contradicts the initial perception of inaccuracies associated with the periodic flow-field assumption.

- Upon thorough deliberation of the above observations in conjunction with the highly improbable development of a uniform velocity profile along the fuel bundle run, it is reasonable to infer that large variances in compared (k) solution values are not a materialization of the periodic flow-field assumption. Rather, the increased congruency between pressure losses from simulations imposing a uniform inlet velocity profile and SNL experimental measurements is likely due to the enhancement of form losses connected with augmented velocity profile development. This inadvertently compensates for actual form losses that are internally fostered elsewhere across the assembly flow segment, which are not properly accounted for by the computational model or flow-field resolution.
- Generally speaking, other notable sources of potential error in (k) term values include but are not limited to deficiencies associated with the domain mesh model, setup as well as execution of the computational solver, solution convergence criteria, and assessment of incomplete iterative convergence error.
- Comprehensive consideration and meticulous discussion of enumerated specifics pertaining to each of these principal topics in formulating the computational experiment framework are conveyed throughout preceding sections of this chapter as well as Chapters 2 and 4 above. The expended efforts to supply detailed characteristics and explanations for features incorporated into the computational modeling process coupled to each particular solution set preclude a need for any substantial reiteration of these aforementioned aspects concerning the present reflection.
- In short, all mesh models are fashioned from high quality grid elements that meet or exceed quality metric values consistent with the user's guide (FLUENT Inc., 2006) for constructing a propitious mesh of requisite cell resolution to capture

anticipated solution gradients and flow-field developments throughout the computational domain. These grid features follow in accordance with suggested stipulations and guidelines from the user's guide (FLUENT Inc., 2006) specific to appropriate numerical model implements for each solver setup.

- Other corresponding points of deserved emphasis relevant to every solver setup and simulation assessment include the utilization of only second-order accurate numerical schemes, selections of significantly tighter residual convergence criteria (for all equations) than suggested by the user's guide (FLUENT Inc., 2006), and establishment of two or more suitable convergence monitors.
- Additionally, the agreement between SNL experimental data and simulation trial results from the bundle run model (periodic boundary conditions) for predicted pressure loss (see Tables 5.6 and 5.7) ultimately heightens confidence in the correctness of associated mesh generation and solver setup facets applied. This is further evident by the similarity among compared (S_{LAM}) values from the SNL hydraulic analysis report (Durbin and Lindgren, 2005) and those obtained for both fully-populated segments of the fuel bundle run and spacer model (see Tables 5.10 through 5.13); the spacer-junction (S_{LAM}) value is also validated in Chapter 6 using a proposed approximation method.
- Although definitive confirmation necessitates continued research efforts accompanied by augmented models of the fuel assembly domain, appositely attributing aspects of current mesh constructs, solver setups, and convergence criteria as the predominant sources of error in (k) term solutions is deemed highly improbable.
- Finally, based on the acknowledgments described above and present assumption of ideal experimental measurements for documented SNL analysis results (Durbin

and Lindgren, 2005), the scope of practical explanation for discrepancies revealed between compared (k) terms is expressly narrowed in focus to a single, logical source of modeling error. Specifically, conflicting (k) term solutions are perceived as a prevailing consequence of the initially accepted hypothesis proposed in Section 4.1, which stipulates domain simplification through the removal of outer grid-strip flow deflection vane and central spring/inner-ring linkage mixing devices from modeled spacer components (see Figures 4.13 and 4.14); assumed to incur only a minute, tolerable increase in resultant error.

- Despite several convincing arguments supporting the preliminary hypothesis statement, sizeable increases in the differences between compared pressure losses that coincide with extension of the modeled domain segment to include a fully-populated spacer element severely undermine the legitimacy of implemented simplifications to flow volume (spacer) geometry assuming negligible propagation of modeling error.
- In fact, close examination of the analysis results not only reveals a large escalation in pressure loss error accompanying the modified (spacer) component addition, but also implies that a vast majority of this error stems solely from inaccurate form loss predictions across the assembly segment. This inference is drawn from the observation of notably contrasting SNL and CFD simulation (k) values and careful consideration of the physical flow losses characterized (or represented) by a (k) term as defined within the curve-fit equation model.
- The general design and function of integrated mixing devices on spacer components improve thermal fluid heat transfer within a fuel assembly operating under normal BWR in-core conditions. However, this improvement emerges at the cost of increased pressure loss across these elements arising predominantly

through heightened flow separation and dissipation phenomena in terms of a quantified minor (or form) head loss imposed by each set of mixing devices.

- Congruency between the observed discrepancies in (k) term solutions and expected consequences resulting from the removal of spacer mixing devices through effected influences on flow behavior strengthens the above explanation for differences in pressure loss comparisons with SNL measurements as well as attributed source of primary modeling error.
- On the other hand, Equation (3.9) is far from representing a complete numerical model of all flow losses, which are characterized by only a major and minor loss term grouping in the fitted equation. As such, this simplified model falls short of providing a precise mathematical description or account of all losses incurred within the fuel assembly and is therefore incapable of explicitly discerning the overall pressure loss contribution arising from the development of specific flow-field features.
- Hence, the observed trend in under predicted values associated with (k) terms derived from simulation trial results cast in the form of Equation (3.9) only infers that deficiencies for minor pressure losses exist as a probable manifestation of flow developments negated by removal of mixing devices from spacer components. More importantly, the possibility of other identified or unidentified developments within the flow domain that solely or jointly prompt this effect on (k) term solutions (as a direct inclusion or indirect exclusion of the initiating source and corresponding flow-field response) is not precluded. In essence, further modeling and analysis efforts are requisite for a more definitive deduction of the specific flow feature(s) implicated in the observed solution trend and underestimated values for segment (k) terms.

- Ultimately, a concluded outcome formulated on findings presented above from the current extent of completed computational research preliminarily confirms the primary source of differences in pressure loss comparisons with SNL measurements does not originate from deficiencies within the FLUENT CFD software, mesh model development, or numerical solver implement. Rather, it originates from a costly, joint-initial decision with SNL researchers to remove the mixing devices located on spacer components from modeled flow domain geometry in order to simplify the mesh generation process, improve grid element quality, and reduce computational resource requirements/costs.
- As a result, foremost recommendations for ongoing research include insertion of mixing device geometries into the meshed flow domain model, completing a numerical accuracy and uncertainty analysis with emphasis on viscous submodel setup after the addition of mixing devices, and then repeating final simulation trials as well as hydraulic loss coefficient evaluations for comparison with SNL experimental results.
- Based on grid generation efforts imparted in developing the mesh models utilized in this research, an estimated minimum of 25 to 50 million additional volume cells are requisite for accurate representation of mixing device surfaces; assuming adequate grid refinement is achievable.
- Furthermore, due to the significant impact observed on downstream pressure losses imposed by flow-field feature developments originating in close proximity to a spacer component (intensified with the insertion of mixing devices), another recommendation is to extend the downstream modeled length of the fuel bundle run following a spacer segment in order to enhance the resolution of these secondary flow losses. In other words, for the current domain model

encompassing a single full-length bundle run and spacer segment, a more suitable placement of the inlet/outlet boundaries is afforded by reasonably decreasing and equivalently increasing the bundle run length upstream and downstream of the spacer component, respectively.

- Referencing flow-field solutions obtained from the completed research above, an appropriately conservative modification for initiating future simulation trials is approximately a 50% reduction and equivalent increase in the current bundle run length defined upstream and downstream of the spacer component, respectively; obviously, the combined total length of the bundle run sections remains unchanged and equal to a full-length segment value.

As previously mentioned, an alternative approach to the consideration of solution discrepancies is based on observations characterized by perspective #2, which presumes a significant fraction of the observed differences among compared results originates from deficiencies in the SNL experiment and hydraulic analysis of measurement data. However, due to the author's lack of direct involvement in SNL experimental trials and a requisite avoidance for introducing any uncorroborated inferences centered largely on speculation, the breadth of observational considerations falling under perspective #2 is somewhat limited in contrast to perspective #1 above. Nevertheless, these few observations stemming from the alternative perspective impart a greater understanding and unique interpretation of the existing solution differences. Thus, for reasons of completeness, the observational points as well as construed explanation(s) concerning notable solution differences (based on perspective #2 and related assumptions) are thoroughly discussed in the bullet-delineated list below.

Key Observations of the Results Analysis from Perspective #2:

- Foremost, prior substantiation of an inherent error associated with the implemented SNL hydraulic analysis approach for deriving major/minor loss coefficient values from experimental pressure measurements of a specific form is central to the focus of ensuing deliberations; this inaccuracy is elucidated in kinetic energy coefficient discussions surrounding Equations (5.30) through (5.32) from Section 5.2.2 above.
- In particular, the sole utilization of differential static pressure measurements by SNL in their hydraulic analysis fails to account (precisely) for the kinetic energy coefficient values dictated in Equation (5.31) or the potential deviations among these terms as necessary for accurately resolving the head losses incurred between assembly pressure ports.
- Unfortunately, definitively quantifying the error introduced into each loss coefficient calculation employing (only) static pressure measurements with regard to all examined segment combinations stipulates extensive access to SNL experimental data sets and additional modeling analysis endeavors that stretch far beyond the scope of current research objectives. Consequently, the following observational assessment and corresponding rationalization for discrepant term solutions are intended as a best-effort attempt to illuminate the evidence indicative of this error source and reasonably refine the probable range of influence on resulting loss coefficient values.
- Although the predominantly inaccurate solution set assumed under perspective #2 pertains to the SNL experiment and hydraulic analysis of measurement data as opposed to the computational model and simulation results under perspective #1, the observed discrepancies between compared solution sets remain unchanged.

Hence, as thoroughly outlined with perspective #1, the notable findings from solution comparisons are generalized by limit-exceeding differences between all contrasted values for (k) terms of equivalent segment definition while conversely exhibiting an acceptable level of agreement between the corresponding (S_{LAM}) term values.

- The onset of observational descriptions uniquely characterizing the prevailing source of above-mentioned error (under assumptions defining perspective #2) is initially marked by a physical interpretation for the general flow loss phenomenon inherent to each group of terms in Equation (5.31) and the plausible direct link with terms expressed and pressure losses represented by Equation (3.9) in equating these two expressions. A particular emphasis is placed on the evaluation of grouped terms containing the kinetic energy coefficient (*i.e.* middle group of terms in both enclosed sets of parentheses) from Equation (5.31) and the anticipated type(s) of resulting pressure loss associated with each group.
- While individually each of these grouped terms quantify the kinetic energy of flow at a cross-sectional plane location with the kinetic energy coefficient correcting for a nonuniform velocity profile development, the differential value between these two groups of terms encompasses a more in-depth meaning. Specifically, a nonzero difference in values indicates that a clear deviation from the fully-developed flow state (or complete lack of) is present between measurement locations described by the two middle groups of terms in Equation (5.31), which springs from an accompanying modification to the kinetic energy of fluid flow along this path segment.
- As thoroughly discussed, the major losses are defined as those due to viscous effects in fully-developed flow through constant geometry areas, which

correspond with the first set of terms cast on the right-hand side of Equation (3.9). Subsequently, the losses characterized by a nonzero differential value between the two grouped terms containing a kinetic energy coefficient in Equation (5.31) that signify a departure from the fully-developed flow state as well as a variation in velocity profile at each measurement point fall outside this classifying description for major losses.

- Therefore, it reasonably follows that such kinetic energy losses are most directly accredited to the second set of terms cast on the right-hand side of Equation (3.9), which represent minor losses incurred due to inertial flow effects stemming from entrances, geometry/area changes, outlets, and other modifying structures of velocity profile development. This formulation ultimately infers that inaccuracies in the evaluation of kinetic energy coefficients (or denoted groups of terms) impart the greatest detriment (error) to solution values derived for (k) hydraulic loss coefficients.
- The observance of sizeable differences between compared (k) term values (for stated assumptions under perspective #2), combined in agreement with considerations of the formulated inference above, significantly strengthen support for ascribing inconsistencies introduced through confined/sole utilization of differential static pressure measurements in the hydraulic analysis conducted by SNL as a principal source of solution discrepancies.
- Accordingly, the associated significance of this attributed error increases dramatically in the vicinity of a spacer component as suggested by pressure gradient trends illuminated in Figure 5.80, which also affords an indirect gauge/scale for qualitatively assessing the error magnitude over a range of specified flow rates.

- Furthermore, other observational constituents providing additional recognition and explanatory support for distinguishing the error contributing source arise directly out of characteristic solution features discerned only from SNL hydraulic analysis results reported by Durbin and Lindgren (2005), which are aptly indicative of (though not exclusive to) construed influences originating from the perceived error source.
- The hydraulic analysis solutions reported by SNL that possess element aspects of greatest supporting interest and least observational ambiguity are addressed in a number of subsequent key points pertaining to the results displayed in Figure 3.2, Table 3.4, Table 3.5, and Table 3.7 from Chapter 3 above.
- Reflecting upon Table 3.7 initially, marked differences in the calculated effective and physically measured coefficients, allied to an equal coefficient value assumption used for identical components in computing the effective set results by Durbin and Lindgren (2005), corroborate a number of expectations regarding flow-field developments through the SNL experimental assembly apparatus.
- Specifically, these differences signify anticipated variations in both the degree and rate of flow-field (velocity profile) developments through the fuel assembly especially near the domain inlet, fully-populated to partially-populated fuel transition, and domain outlet; even if multiple-component segment extents are located at the same relative position within identical upstream and downstream components, the potential for velocity profile variations still exists.
- Moreover, it follows from this line of reasoning along with preceding explanations for observed comparison discrepancies that inadequate application of the equal coefficient value assumption (for identical components) results in heightened divergence between the measured and effective coefficient sets for

segments encompassing flow-field locations of significantly varying velocity profile development. This error difference is notably greater for the (k) terms in general and emphasized particularly among short multiple-component or single-component segment spans with upstream extents immediately preceding a largely undeveloped (or redeveloping) velocity flow field.

- For example, pressure drop segments (2-8) and (4-8) in Table 3.7 are most congruent with this description and therefore warrant an expectation of higher discrepancy among the measured and effective coefficient values, especially concerning the compared (k) terms. These characteristics are clearly revealed within Table 3.7 upon a brief comparison of the presented coefficient values.
- In terms of the detrimental influence to loss coefficient values based directly on actual SNL experimental measurements (the measured values in Table 3.7) using only static differential pressures, the above-mentioned concerns remain widely applicable, but are extended with even greater focus toward the smaller flow segments of one or (at most) two adjacent assembly components. These segments in which diverse developments exist in the flow-field behavior, velocity profile, and kinetic energy of airflow between the segment extents (or bounding pressure ports), not precisely accounted for by differential static pressure measurements alone, also absorb considerable inaccuracies into the resulting calculations for loss coefficient solutions.
- The inaccuracies introduced are most evident through unphysical and/or abnormally large variations in (k) term values for single- or multiple-component segments located adjacent or in successive order to one another, which possess identical cross-sectional component geometries.

- For instance, several segment (k) terms of exactly (or numerically approximate) zero magnitude are explicitly reported or indirectly derived from the measured (k) loss coefficient solutions in Tables 3.5 and 3.7 above, which principally describe unphysical flow-field losses across these segments. Such assembly span lengths of noteworthy interest include (but are not limited to) pressure drop segments (5-6) and (7-8) from the partially-populated fuel region with (k) term values based on the respective bundle run flow area for this assembly section.
- The (k) loss coefficient value of 0.00 is visibly attained for pressure drop segment (5-6) from Table 3.5 directly. However, the (k) term for pressure drop segment (7-8) is indirectly obtained through the measured (k) coefficients listed in Tables 3.5 and 3.7 by deducting the aggregate (k) quantity for segments (4-6) and (6-7) from the encompassing segment (4-8). This calculation reveals a (k) value of 0.02 and near negligible magnitude for segment (7-8) form losses.
- Alternatively, measured (k) coefficients listed in Tables 3.5 and 3.7 employing the fully-populated bundle run flow area as a hydraulic geometry basis provide a second avenue for determining the (k) term value for pressure drop segment (7-8) similar to the preceding partially-populated formulation. The only additional calculation necessary in this approach is an initial assessment for the unspecified (k) coefficient of segment (4-8) assuming a fully-populated bundle flow area, which is simply acquired by resolving the difference between (k) term values of segments (4-15) and (8-15) as listed in the tabulated results.
- Subsequently, by following the same calculation procedure as outlined previously for establishing (k) based on the partially-populated bundle flow area, an alternative (k) coefficient value of -0.31 is also ascertained for segment (7-8)

using the fully-populated bundle flow area and corresponding measured (k) coefficient data from Tables 3.5 and 3.7 in a second computation.

- Although both (k) loss coefficient values of 0.02 and -0.31 assuming partially- and fully-populated bundle flow areas, respectively, for segment (7-8) are potentially susceptible to insufficient significant digits in measured coefficient data reported by SNL (see Tables 3.5 and 3.7), these results stem from two separate calculations and yet congruently support the same general finding of an unphysical (k) value.
- Furthermore, a brief comparison of the (k) term values for adjacent bundle run segments (4-6) and (7-8) from the partially-populated fuel region reveals a large discrepancy in the measured (k) coefficient results for compared values based on both the partially- and fully-populated bundle flow areas displayed in Table 3.5 above. Obviously, these result inconsistencies are just one example segment pair with repercussions that extend back to the SNL assumption of equal loss coefficient values for identical assembly components in computing the calculated effective loss coefficients of multiple-component segments, the results of which are compared in Table 3.7 as previously discussed.
- A similar line of rationalization concerning static pressure loss measurements without specific consideration for kinetic energy variations between measurement points, originating from dissimilar flow developments involving nonuniform velocity profiles, also follows as reasonable explanation for slightly poor fits to SNL experimental pressure loss data in Figure 3.2 at lower end velocities.
- The influential effects are echoed again in Table 3.4, which clearly illustrates notable deviations in the hydraulic loss coefficient values ascribed to fully-populated, partially-populated, and combined assembly fuel regions over different

ranges of experimental airflow rate, especially for initial spans encompassing only the lower airflow rates. Although variations in these loss coefficients are undoubtedly expected with increasing data measurements, flow rate, and range of values considered, the sizeable differences at lower flow rate ranges and remaining divergence evident between even the two largest ranges of airflow rate are particularly concerning.

- Moreover, despite a general trend of increasing (S_{LAM}) and decreasing (k) loss coefficient values with increasing ranges of flow rate for all three fuel regions in Table 3.4 above, the specific differences between (S_{LAM}) values accompanying each of these incremental range increases are not consistently increasing or decreasing across the entire data set of ranges for segments (2-8) and (2-17) examined. As a result, attributing the observed problematic characteristics of loss coefficient values listed in Table 3.4 to experimental measurement influences of a systematic nature is inconclusive at best.
- Unfortunately, the current extent of completed computational research and experimental data analysis (of SNL report) is insufficient in order to definitively associate the above-mentioned theoretical error indicators with actual observational findings derived from SNL pressure loss measurements without further computational modeling and supporting experimental analysis efforts, which are also requisite for assessing the magnitude of related error with greater precision.
- However, the degree of error influence on resulting loss coefficient values due to kinetic energy differences between pressure measurement ports is curtailed by implementing several key aspects of uppermost consideration in selecting

pressure port locations along the fuel assembly length as well as hydraulic analysis approach employed for deriving segment loss coefficients.

- Obviously, the potential error introduced into loss term calculations is minimized through high-resolution cross-sectional measurements of only MWA total pressure for computing differential pressure losses across any single- or multiple-component segment. In particular, obtaining accurate values for segments spanning the length of a single assembly component or less specifically necessitates differential pressure losses based on MWA total pressure measurements with sufficiently resolved velocity profiles since any two locations along the length of a single component inevitably differ (assuming segment constitutes a logical length of the component) in profile development.
- On the other hand, if differential pressure losses across ports are solely limited to static pressure measurements, loss coefficients calculated for multiple-component segments comprised of a single bundle run and spacer component with pressure measurement ports at periodic locations of expected near-equivalent velocity profile developments (*i.e.* possibly bundle run midpoint) afford sufficiently accurate representations of the incurred flow losses.
- Under the same constraints, loss coefficients for multiple-component segments comprised of more than a single bundle run and spacer component (but equal number of each) are permissible without necessarily introducing any additional error, which reduces the number of calculations required for determining pressure losses over longer assembly spans of periodically repeating component geometry. Again, for such combination segments, including a bundle run and/or spacer component at any segment extent adjacent to a distinctly divergent flow area (*i.e.* fully- to partially-populated fuel transition) or existing profile dependence on

upstream/downstream influences outside the multiple-component span considered (*i.e.* fuel assembly inlet/outlet) is prone to extensive error.

- The best combination of components in a multiple-component segment covering a particular fuel assembly section is discernible through goodness of fit comparisons for different pressure loss data sets stemming from various component groupings.
- Finally, for components adjacent to the fuel region transition or assembly inlet/outlet, an exception to the above-stated guidelines is notable for static pressure-limited experiments in which the isolation of kinetic energy inaccuracies to a single-component rather than a multiple-component segment often reduces the overall error in represented assembly pressure losses. This error reduction is especially prevalent in a majority of the observed data fits to pressure losses at lower and upper ends of the airflow velocity range, but not universally observed in all experimental setup conditions or trial scenarios.

In conclusion, the discrepancies between compared hydraulic loss coefficient values from loop 4 velocity simulation trials and the SNL hydraulic analysis report (Durbin and Lindgren, 2005) almost certainly arise as a partial consequence of distinct inaccuracies associated with the evaluated results in both computational and SNL physical experiments undertaken. Although further modeling and supporting experimental analysis efforts are necessary in order to clarify the magnitude of error contributed by each deliberated source, the observations outlined above from perspectives representing extreme extents of the consideration spectrum impart a greater understanding and unique interpretation of the existing solution differences. Foremost, these discussions and observational findings divulge invaluable modeling/resource

characteristics and solution response aspects that are integral in the progression of simulating and validating a large production-type problem involving a prototypic GNF 9x9 BWR fuel assembly lacking any prior experimental or numerical research data. Additionally, these computational and experimental discrepancies provide explicit direction for continuing research with a strong recommendation that immediate future efforts concentrate directly on the resolution of these issues prior to undertaking a subsequent larger-scale modeling phase of the assembly flow domain.

5.3 SIMPLIFIED REPRESENTATION OF EQUIVALENT FUEL ASSEMBLY

The final phase of completed computational research focuses on the potential application of a simplified geometric model characterized as an equivalent hydraulic length-scale representation of the entire 9x9 BWR fuel assembly between upper and lower tie plates. Although numerical experiments conducted using CFD models discussed in Sections 5.1 and 5.2 addressed numerous aspects of the preliminary computational modeling process for simulating the large production-type problem described, the analyses fail to consider several other meaningful concerns surrounding the basics (nature and structure) of an acceptable modeling procedure/method for this form of flow domain problem. In particular, establishing generalized flow-field solutions inevitably relies upon influences imposed by a few prevalent thermal hydraulic features at work within the physical flow domain (not yet explicitly considered) that distinctly affect the (fuel assembly) system response.

Substantiating the practicality/practicability of further research firmly warrants an investigational study with certain regard to the hydraulic modeling approach proposed by SNL (Durbin and Lindgren, 2005) as well as other anticipatory innovations of parallel

development for resolving the expressed flow fields under complete LOCA conditions through simplified means. Facets of notable importance concerning the direct/indirect characterization of both unheated and heated fuel assembly airflows include (but are not necessarily limited to) the following: (1) adequacy of governing equation(s) to accurately reflect as well as replicate system hydraulic losses interrelated with the momentum source balance driving internal flow across domain; (2) feasibility of obtaining realistic measurement values for requisite parameter inputs defined in governing flow equation(s) via physical experiments or CFD model simulations; (3) effect of distinguishing deviations between unheated, pressure-driven and heated, buoyancy-driven airflow features on predicting the nature/behavior of typical flow-field responses due to contrasting attributes of principal, thermal fluid phenomena; and (4) adaptability of unheated, pressure-driven flow correlation(s) and corresponding factor/parameter values for application in heated, buoyancy-driven model constructs. As such, these elements embody pivotal research objectives for evaluating the viability, range of function, and application costs accompanying a potential, simplified numerical-analysis development as well as prohibitive stipulations coupled to input requirements limiting the overall benefit afforded by such a buoyancy-driven flow implementation.

Successfully accomplishing the above objectives, however, necessitates for comparative analysis the ascertainment of both unheated, pressure-driven and heated, buoyancy-driven flow-field solutions for approximately an entire fuel assembly domain due to conditional dependencies between flow resistance and driving buoyant force, which are further contingent upon developing air property temperatures throughout the flow domain volume. Unfortunately, as concluded in Section 5.2, additional modeling developments and mesh assessments are requisite beyond the current research

achievements prior to establishing the availability of a large production-type mesh model representing a near-complete fuel assembly domain.

Therefore, the prohibitive factors that preclude achievement of listed pivotal objectives eventually all funnel down to a single paramount requisite, which is a computational domain model that accurately reflects the complete 9x9 BWR fuel assembly and necessitates only a manageable computational resources cost. Accordingly, the only plausible solution is the development of a simple, highly-symmetric flow geometry defined by equivalent fuel assembly values for hydraulic dimensions present in the governing curve-fit equation describing pressure loss. Secondly, in order for the simple equivalent flow domain to actual provide any meaningful benefit, the pressure loss response function for a specified range of forced airflow rates must exhibit a certain level of agreement in comparison to the same response function of the actual fuel assembly geometry.

If both descriptions are effectively confirmed, a significant qualitative relationship between observations from comparisons of unheated and heated fuel assembly simulation solutions using the equivalent fuel assembly model and those that would be observed if the actual fuel assembly had been used, is an appropriately supported expectation. As such, meaningful inferences are obtainable from these comparisons that progressively afford insights into additional characterizations of modeling approaches and procedures such as the viability of simplified numerical-analysis developments for predicting assembly response functions using constitutive relationships and empirical data.

In summary, the remaining sections of this chapter focus upon development of an equivalent hydraulic scale model representing the GNF 9x9 BWR fuel assembly with greatly simplified geometry construction and efforts confirming congruency between pressure loss response function of new model and actual SNL experimental assembly

using presented values from the SNL hydraulic analysis report (Durbin and Lindgren, 2005). Moreover, aside from a moderate reformulation of the curve-fit equation used in prior analyses to account for temperature dependent properties in heated simulation trials below, the hydraulic analysis and evaluation of simulation flow-field solutions for presenting comparisons of results remain unchanged throughout chapter end. New research contributions requiring explanation in the remainder of this chapter are almost completely limited to the graphical display, numerical tabulation, and labeling nomenclature with description. Primarily, these concern only model geometry and mesh developments, simulation specifications, and simulation result comparisons. As a result, this is the predominant form and corresponding appearance of chapter remainder.

5.3.1 Initial Considerations

The equivalent fuel assembly model components considered in this work are labeled and described in Figure 5.105 below. As shown, the assembly structure is comprised of eight fuel bundle runs and seven grid spacers. Fuel bundle runs and grid spacers 1 through 5 constitute the fully-populated fuel region of the assembly, while fuel bundle runs 6 through 8 as well as grid spacers 6 and 7 constitute the partially-populated fuel region of the assembly. Thus, four unique geometry cross-sections exist across the represented flow domain of the equivalent fuel assembly model.

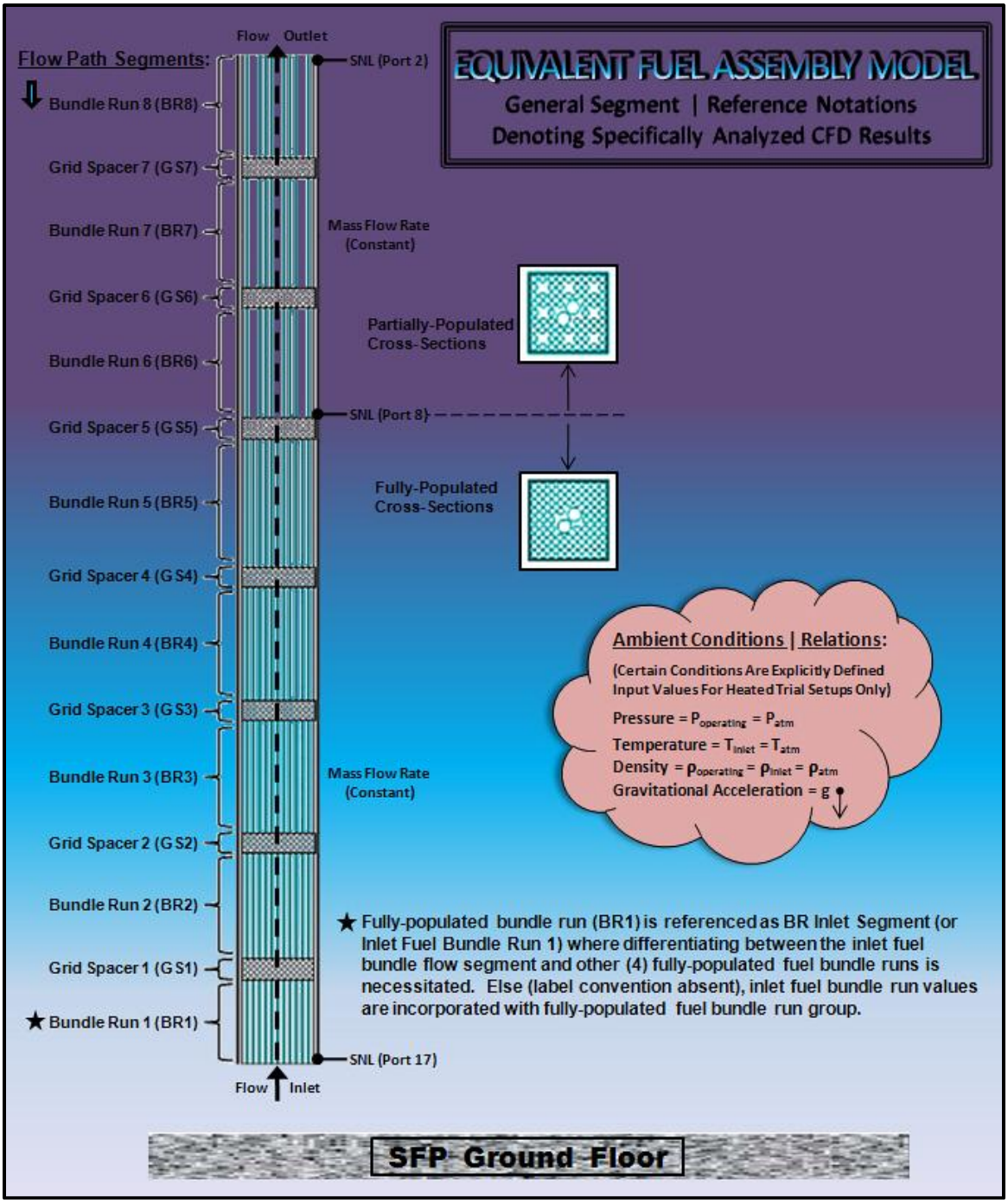


Figure 5.105: General flow segment and domain references for the equivalent fuel assembly model including specific notations utilized in presenting the simulation trial results.

Figures 5.106 through 5.110 below detail the basic design premise and fundamental purpose/objective of the equivalent hydraulic geometry development underlying general construction of the model flow domain including relevant assumptions, defining equations, and an alternative domain representation of functional equivalence via partially-populated region translation.

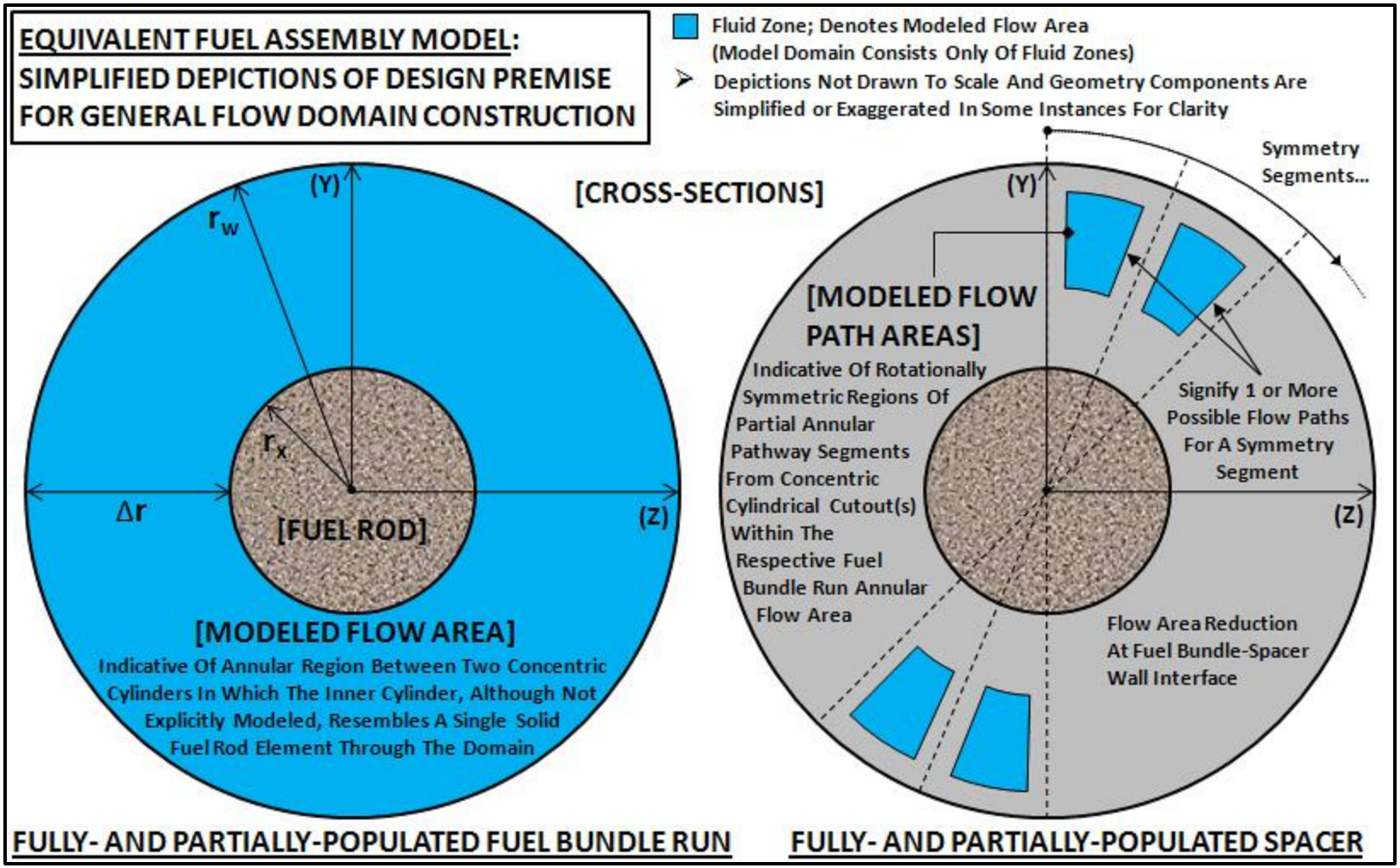


Figure 5.106:Simplified depictions describing general domain construction of the equivalent fuel assembly model (1 of 5).

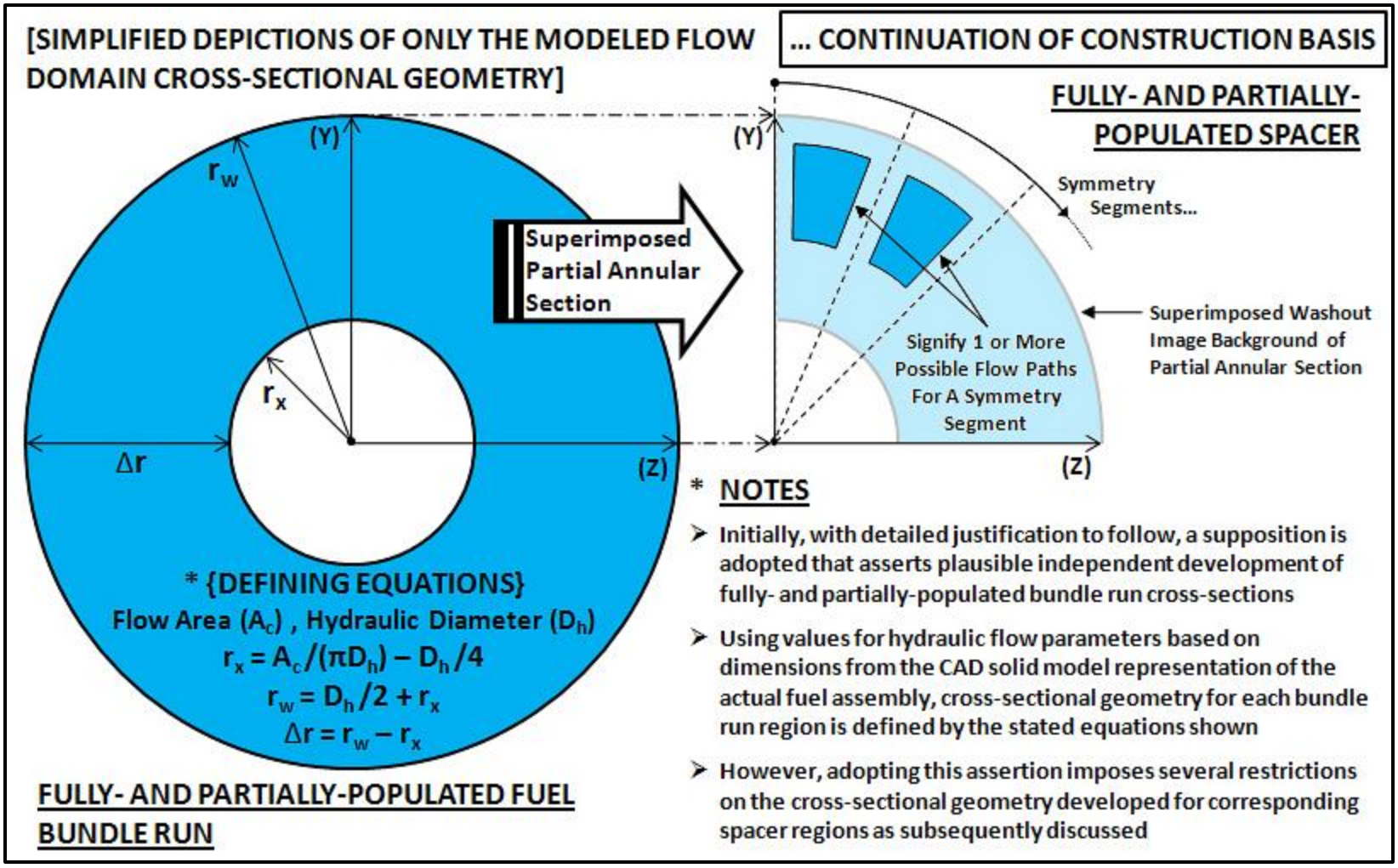


Figure 5.107:Simplified depictions describing general domain construction of the equivalent fuel assembly model (2 of 5).

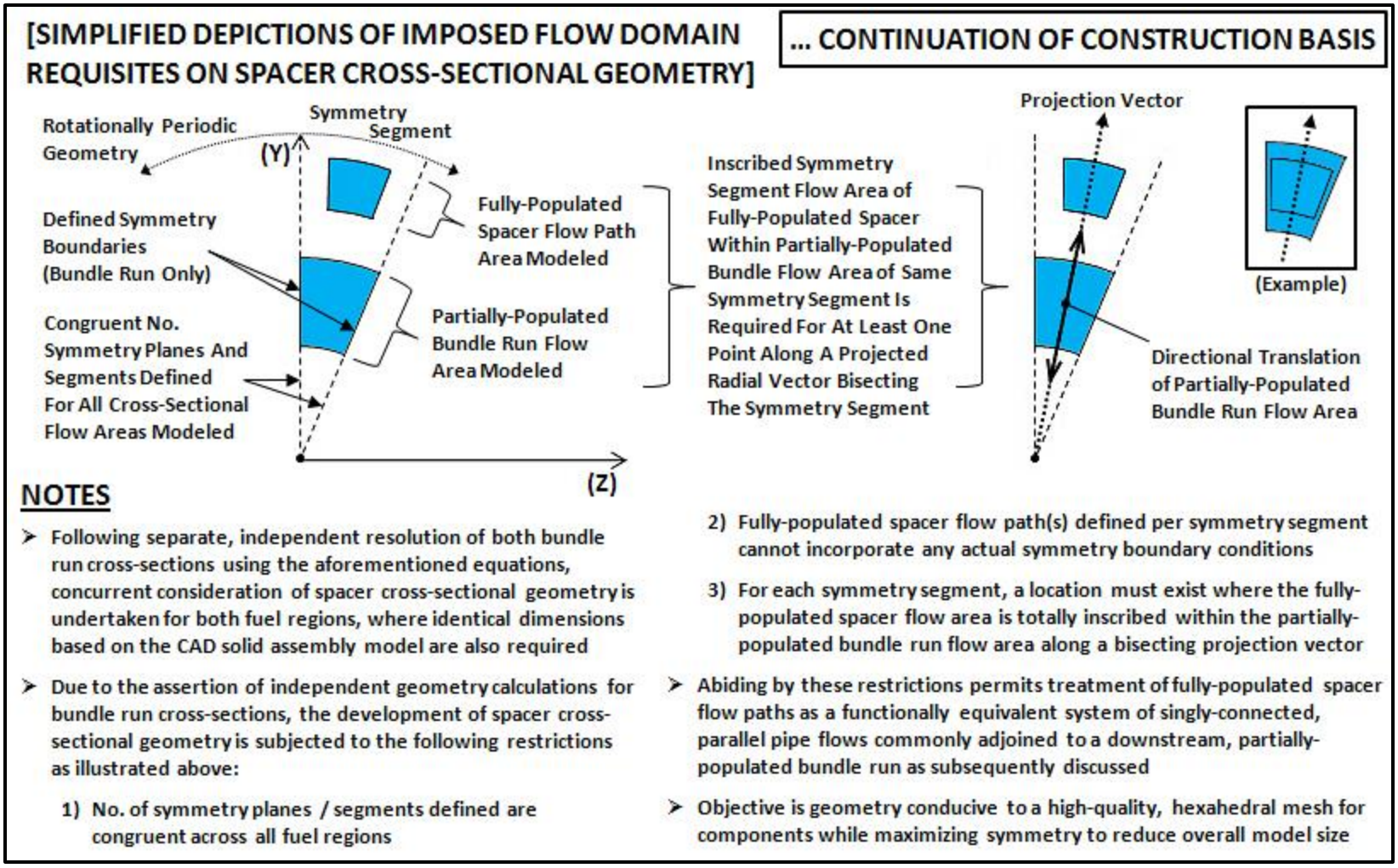


Figure 5.108:Simplified depictions describing general domain construction of the equivalent fuel assembly model (3 of 5).

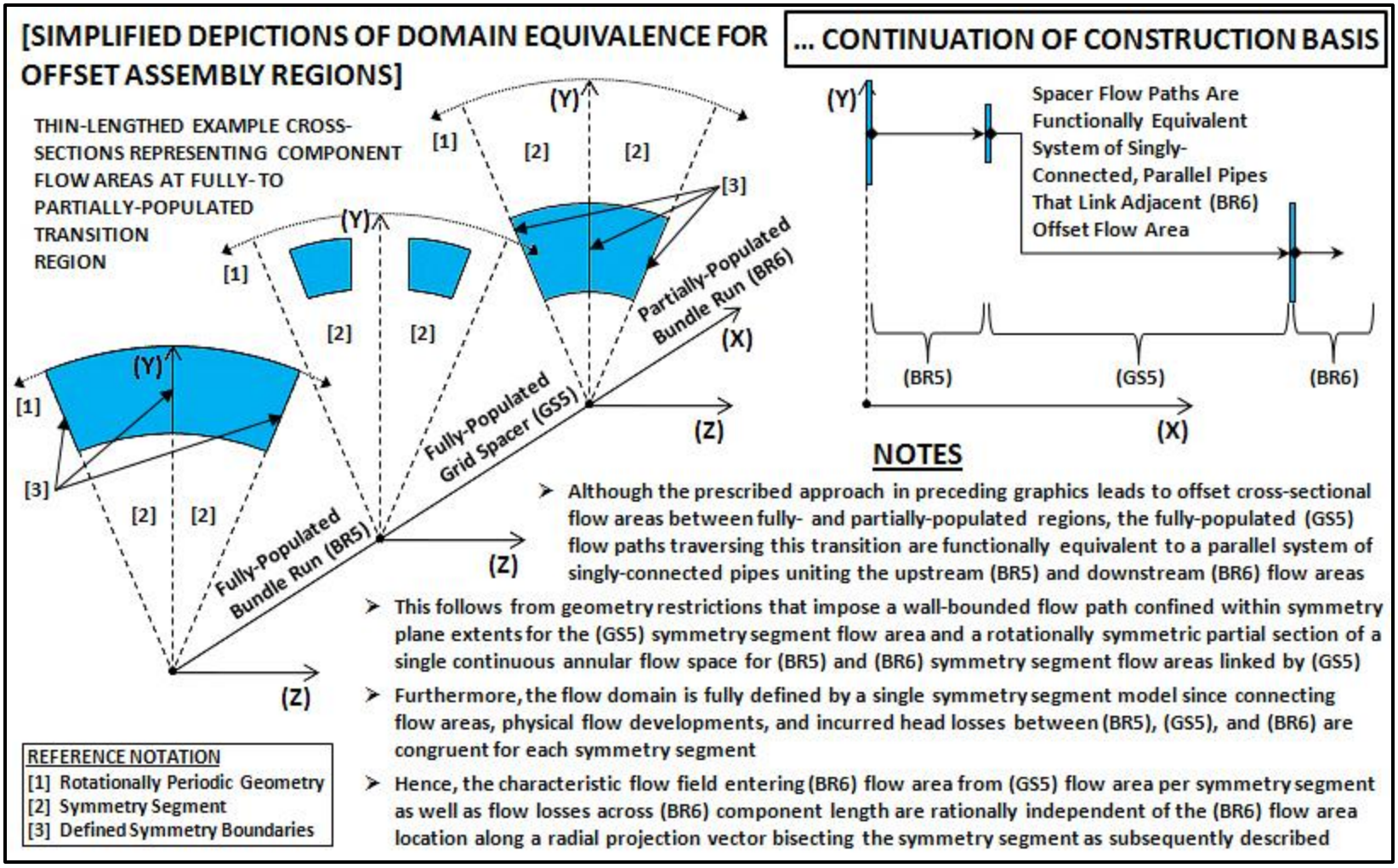


Figure 5.109:Simplified depictions describing general domain construction of the equivalent fuel assembly model (4 of 5).

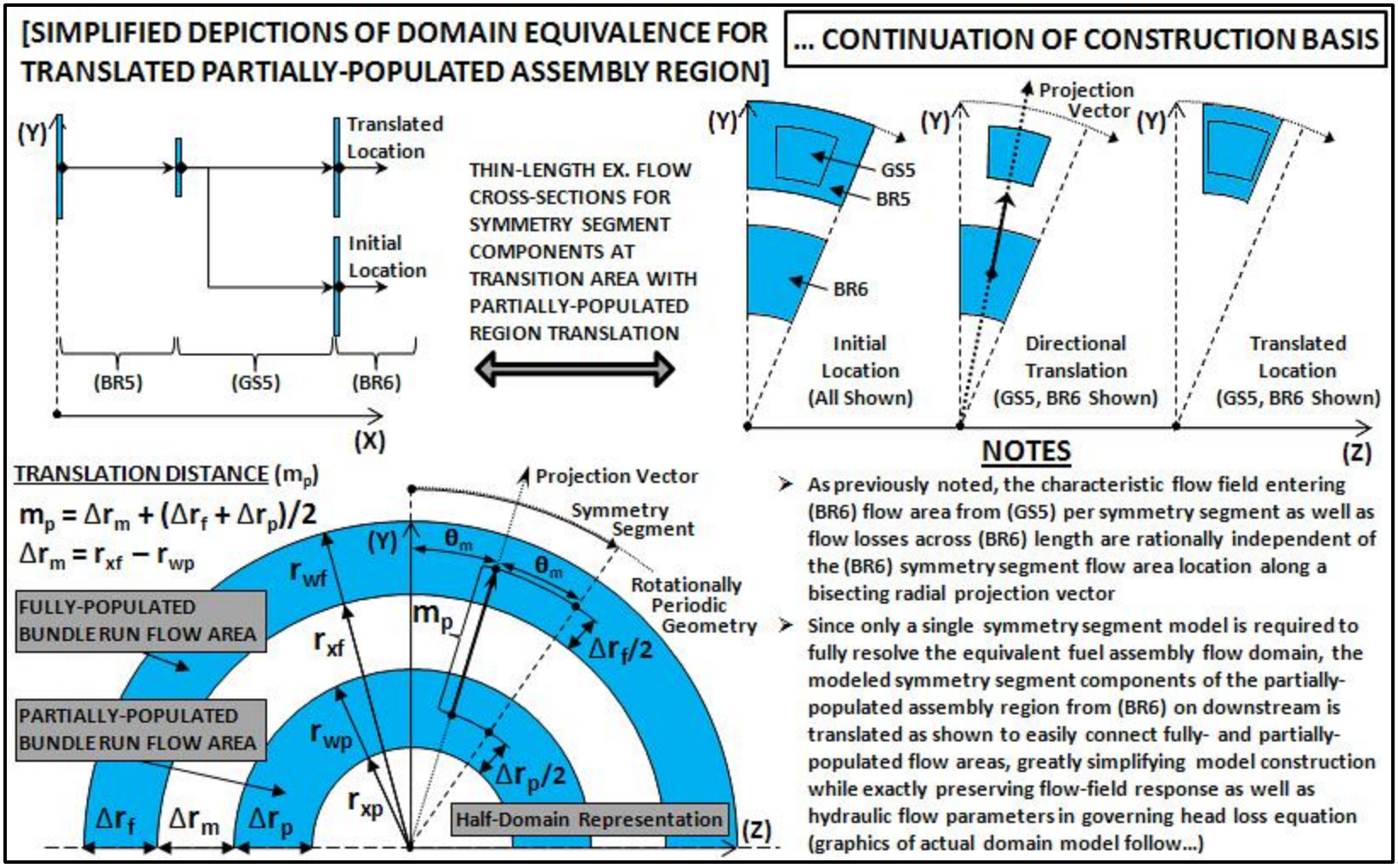


Figure 5.110:Simplified depictions describing general domain construction of the equivalent fuel assembly model (5 of 5).

In addition, Figures 5.111 through 5.118 below provide an overview of the characteristics concerning actual domain geometry developed with regard to each unique component flow segment including the defined boundary types represented within a single symmetry segment model. A description of each unique interface between adjacent component segments is provided with corresponding depictions as well as delineation of pre- and post-translation interface images at the fuel region transition.

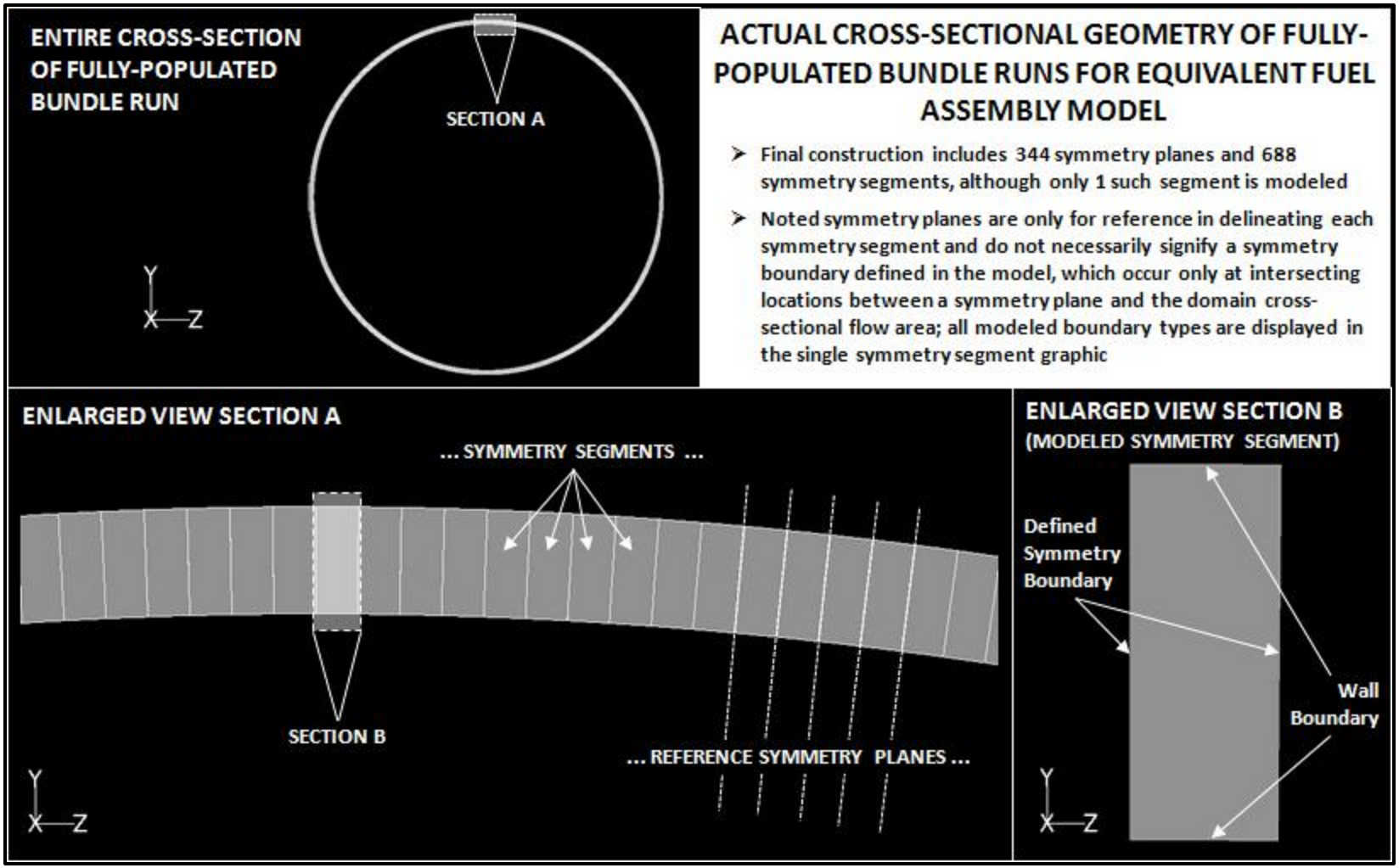


Figure 5.111:Actual cross-sectional geometry of fully-populated bundle run flow area for equivalent fuel assembly model.

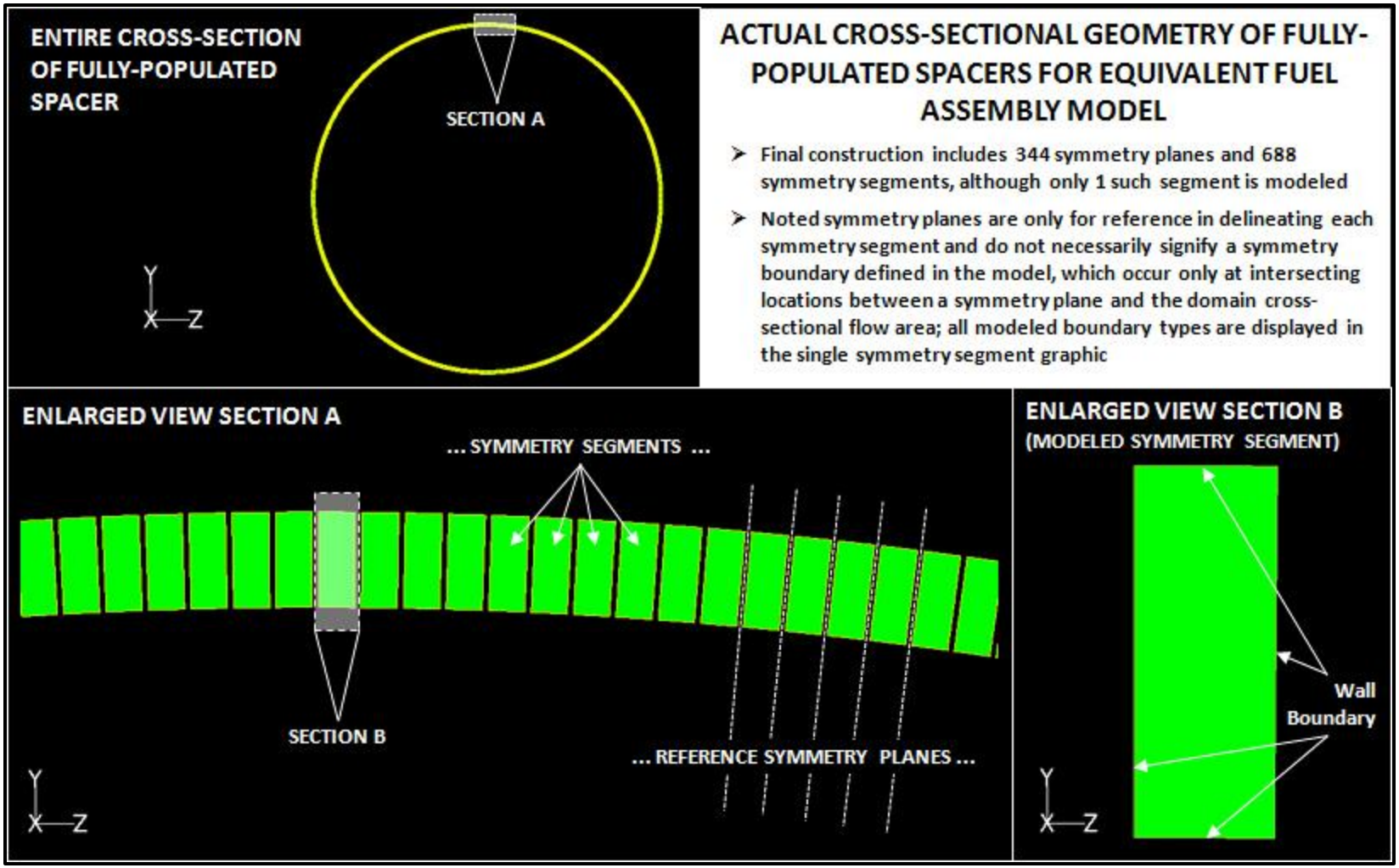


Figure 5.112:Actual cross-sectional geometry of fully-populated spacer flow area for equivalent fuel assembly model.

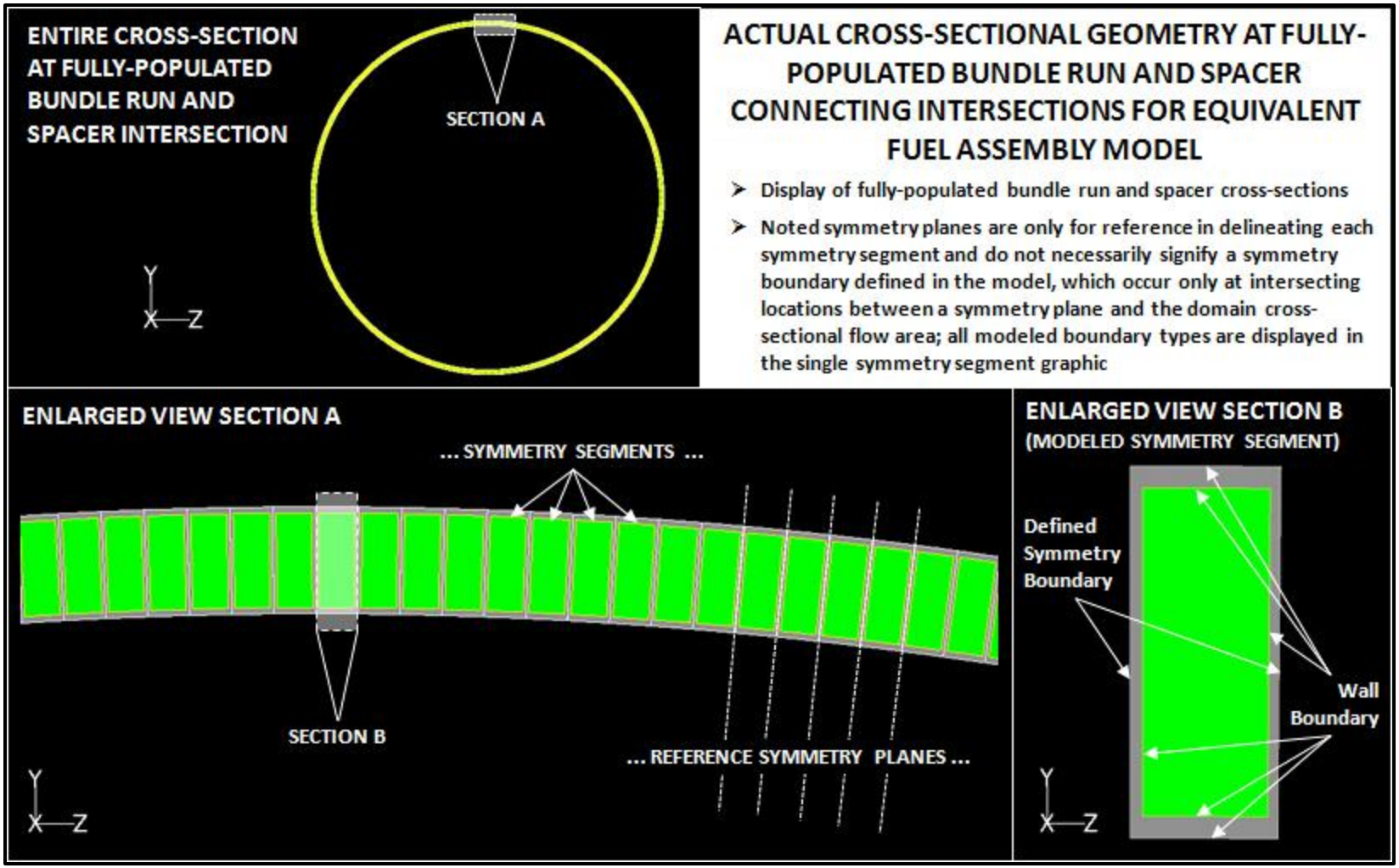


Figure 5.113:Actual fully-populated bundle run and spacer intersection geometry for equivalent fuel assembly model.

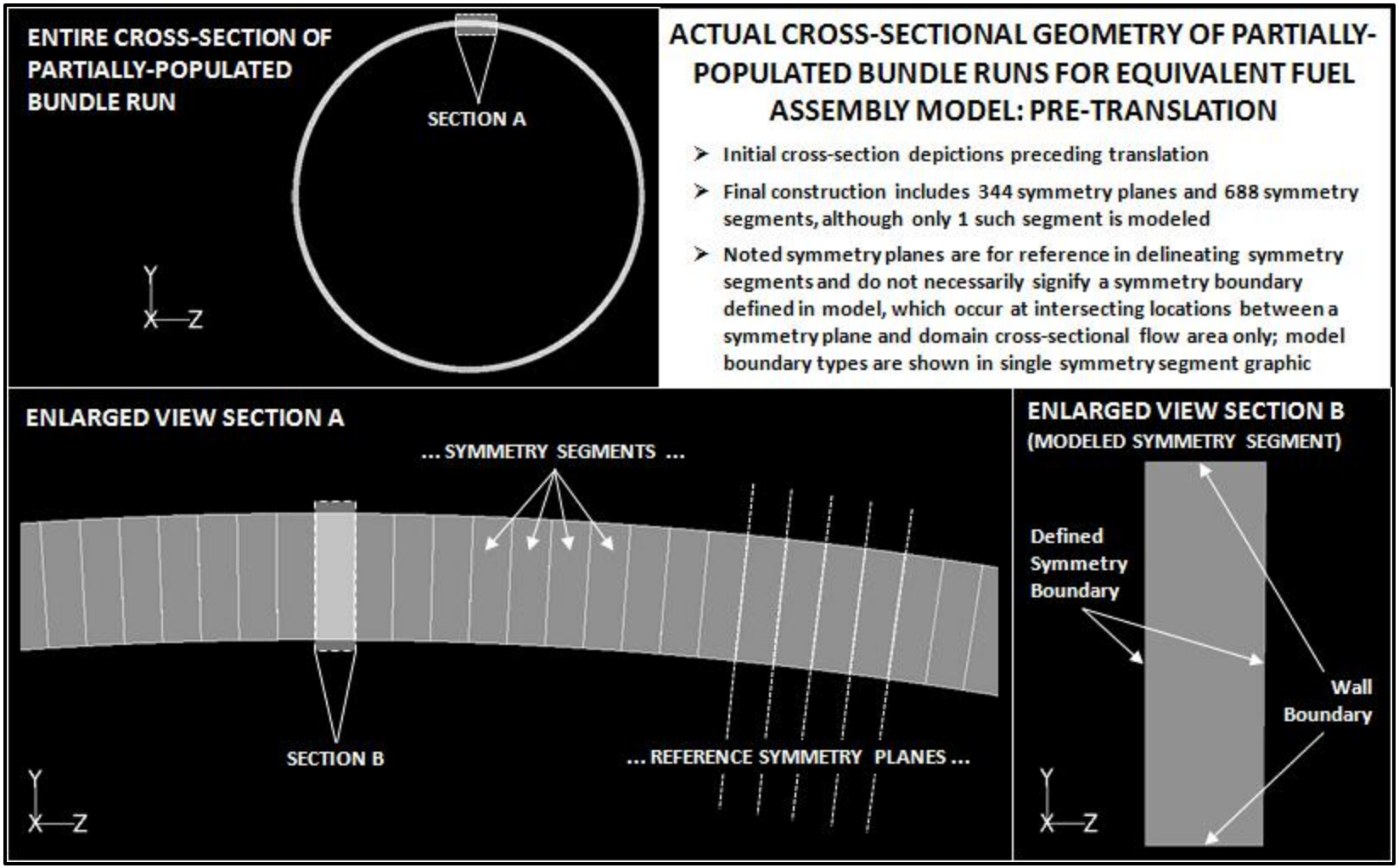


Figure 5.114:Actual cross-sectional geometry of partially-populated bundle run flow area for equivalent fuel assembly model.

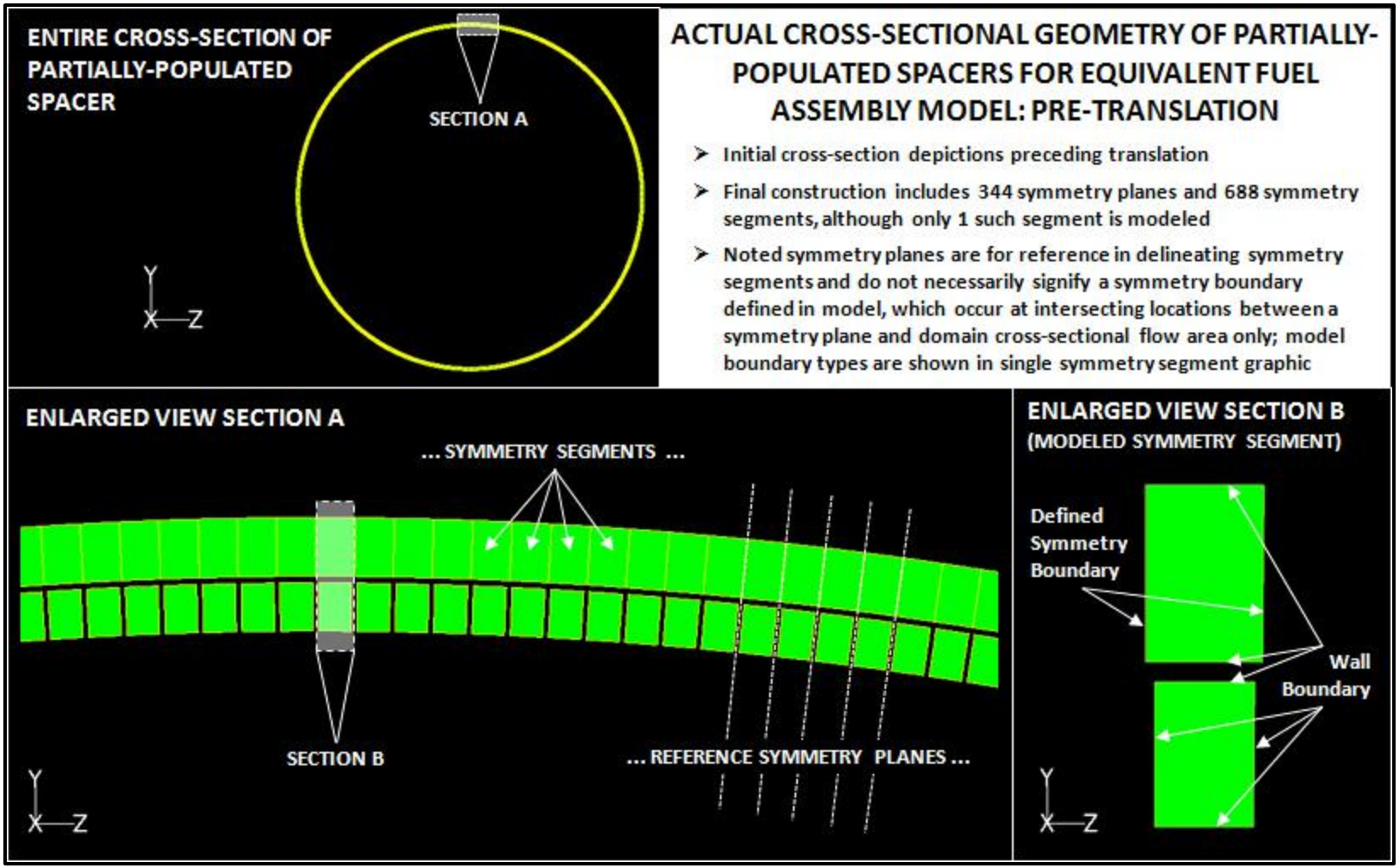


Figure 5.115:Actual cross-sectional geometry of partially-populated spacer flow area for equivalent fuel assembly model.

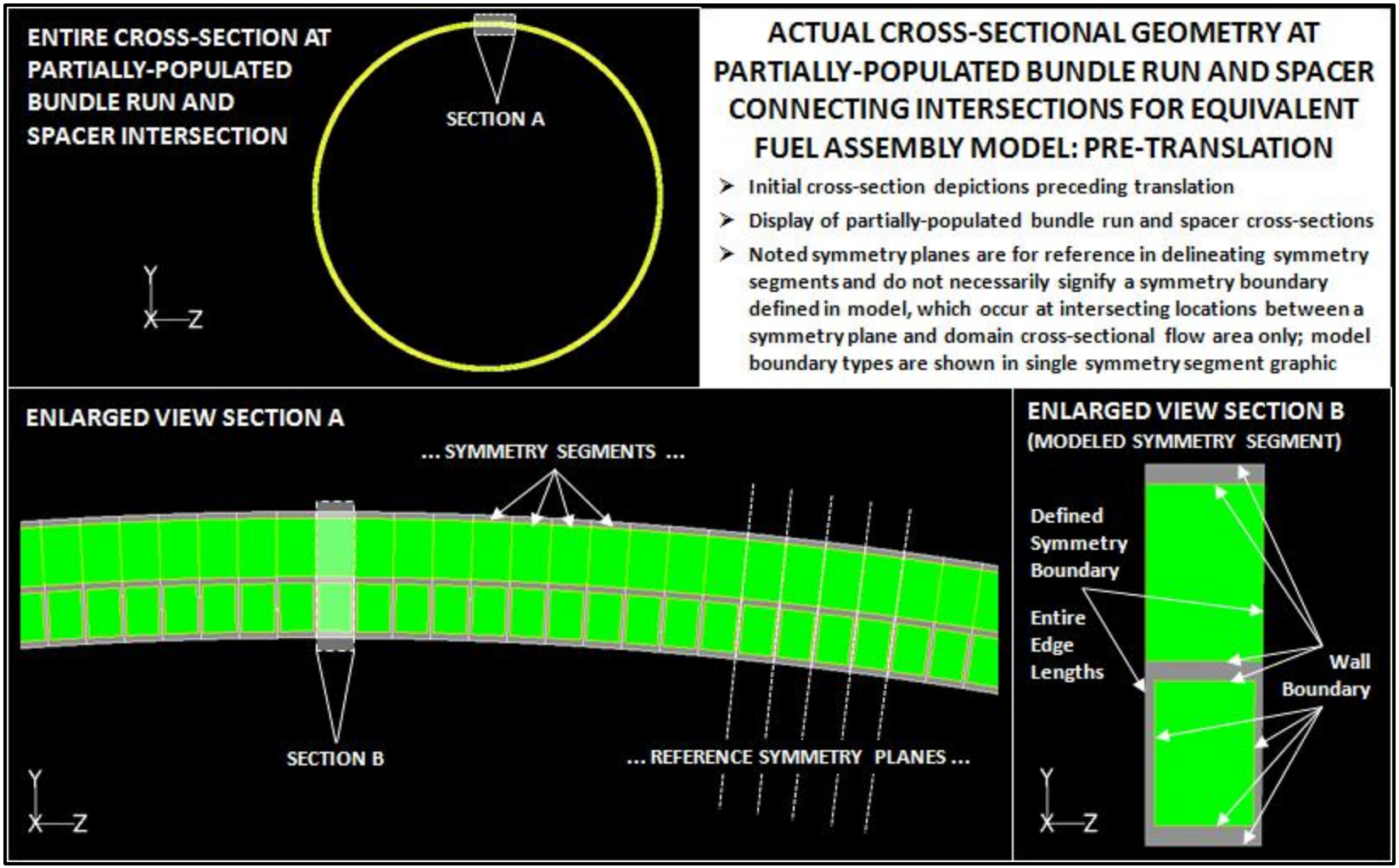


Figure 5.116:Actual partially-populated bundle run and spacer intersection geometry for equivalent fuel assembly model.

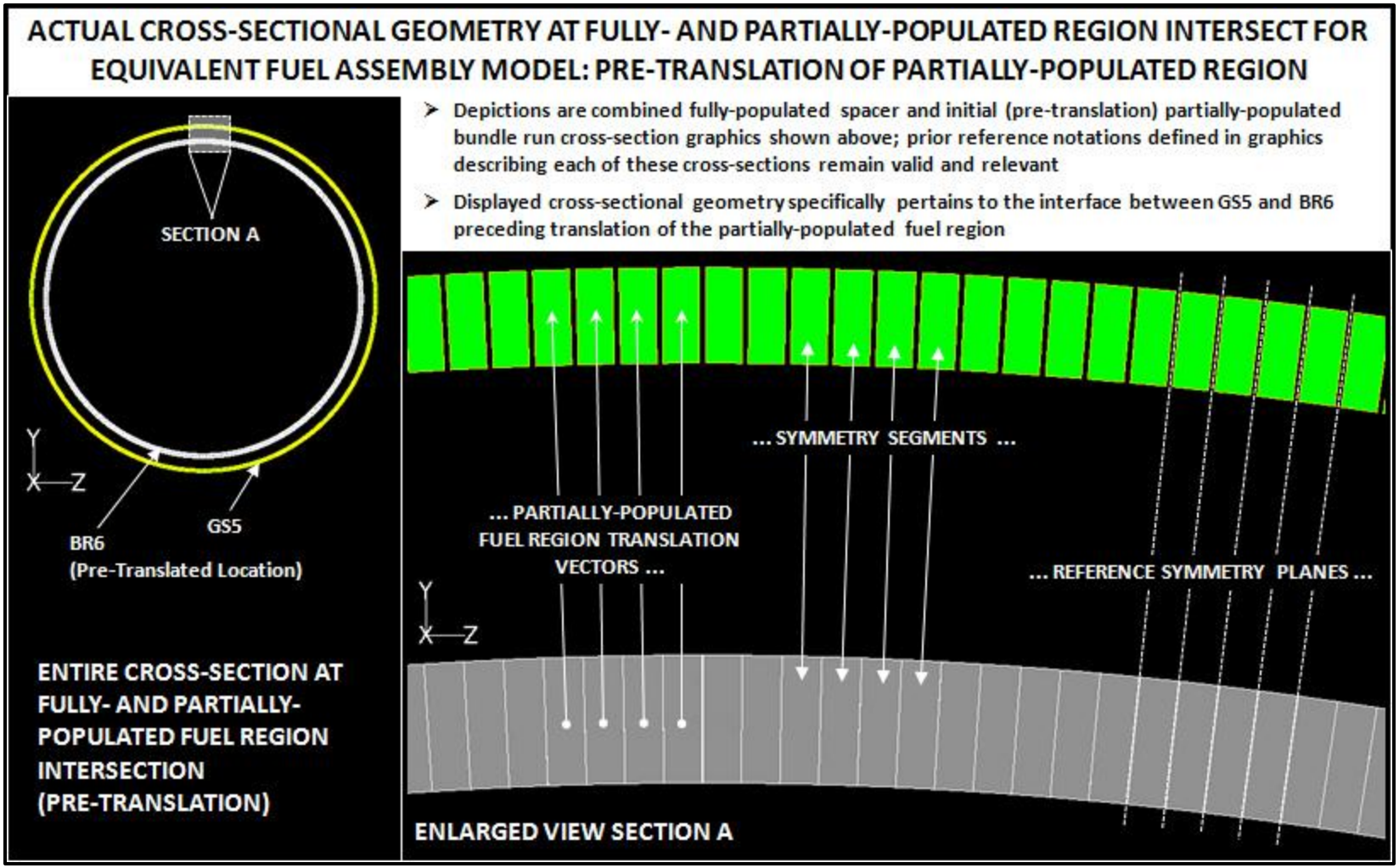


Figure 5.117:Actual fully- and partially-populated pre-translated region intersect geometry for equivalent fuel assembly model.

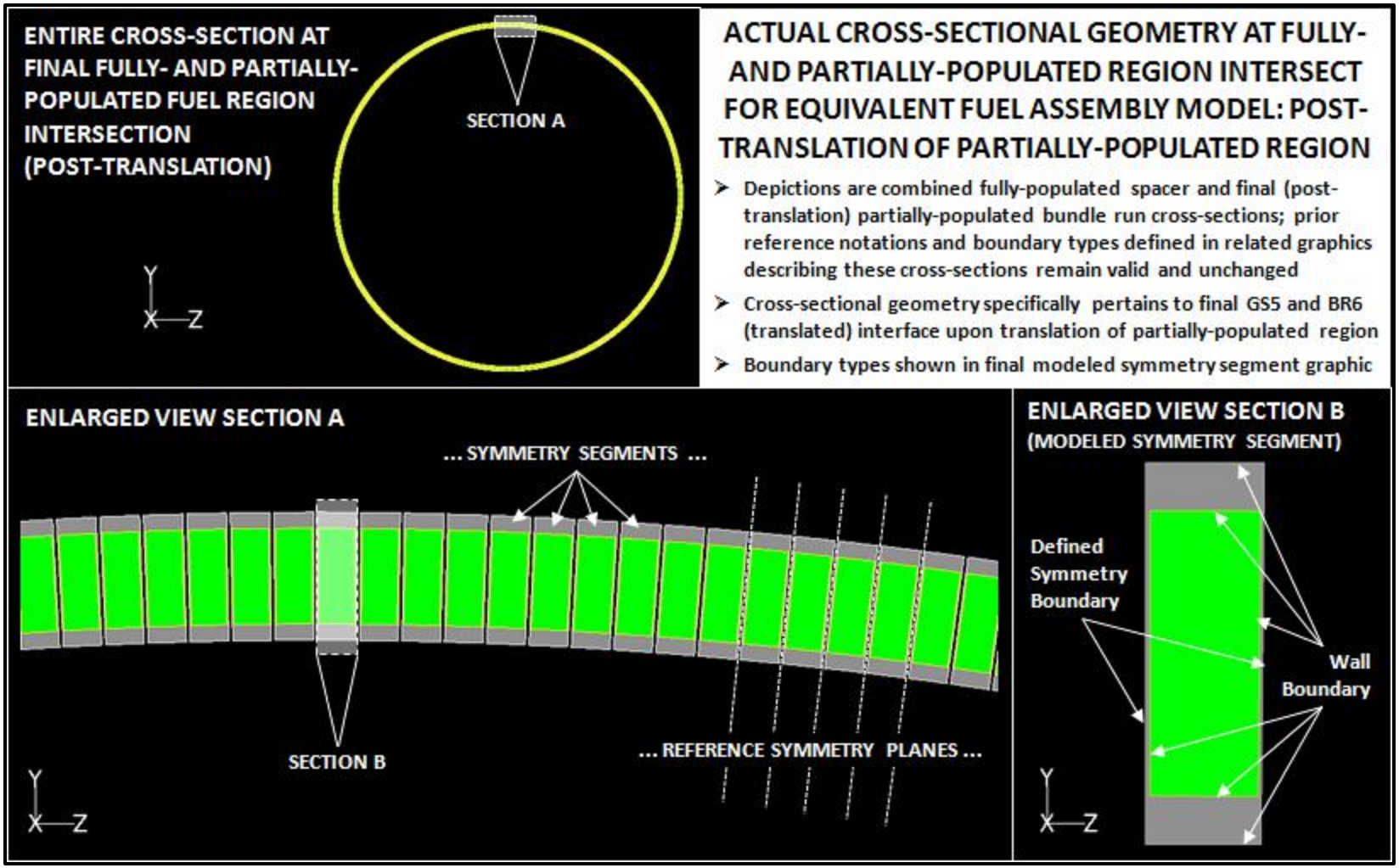


Figure 5.118:Actual fully- and partially-populated translated region intersect geometry for equivalent fuel assembly model.

Figures 5.119 through 5.127 below provide a comparison between cross-sectional flow geometry concerning a single symmetry segment of each unique assembly component as represented within the original CAD solid and present equivalent fuel assembly models, which are both characterized by congruent cross-section (entirety) values for geometry-based hydraulic flow parameters appearing in the governing curve-fit equation. A description of the applied meshing scheme throughout the equivalent fuel assembly model domain is also included among the ensuing graphics. These show the level of simplification utilized in reducing the computational expense of the corresponding simulation runs implementing this equivalent assembly model development. In short, the following depictions consider the comparison of flow domain geometry from both model constructs, stated equivalence between cross-sectional representations, and applied meshing scheme across the computational flow volume.

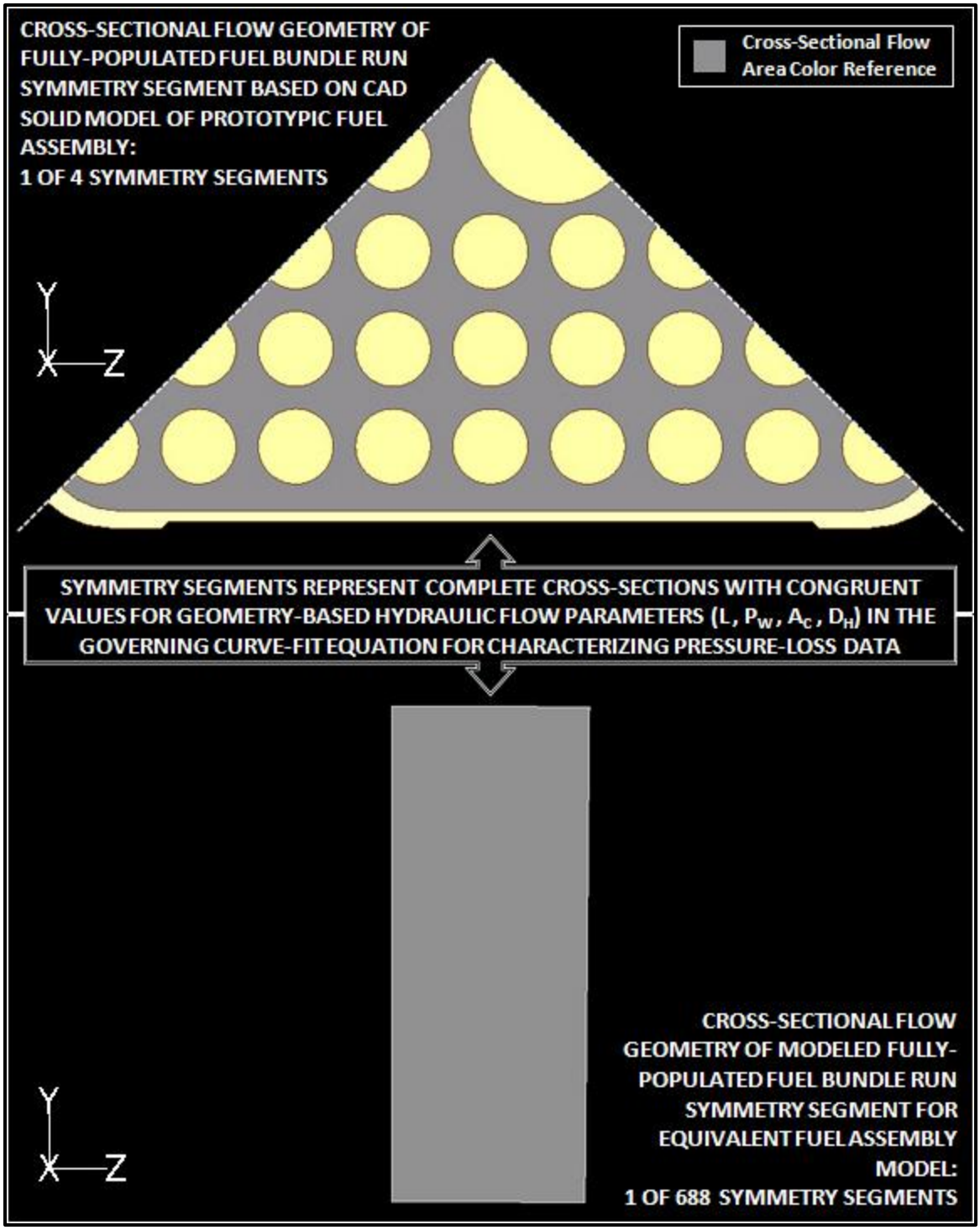


Figure 5.119:Cross-sectional flow domain comparison of fully-populated bundle run symmetry segments from CAD solid and equivalent fuel assembly models.

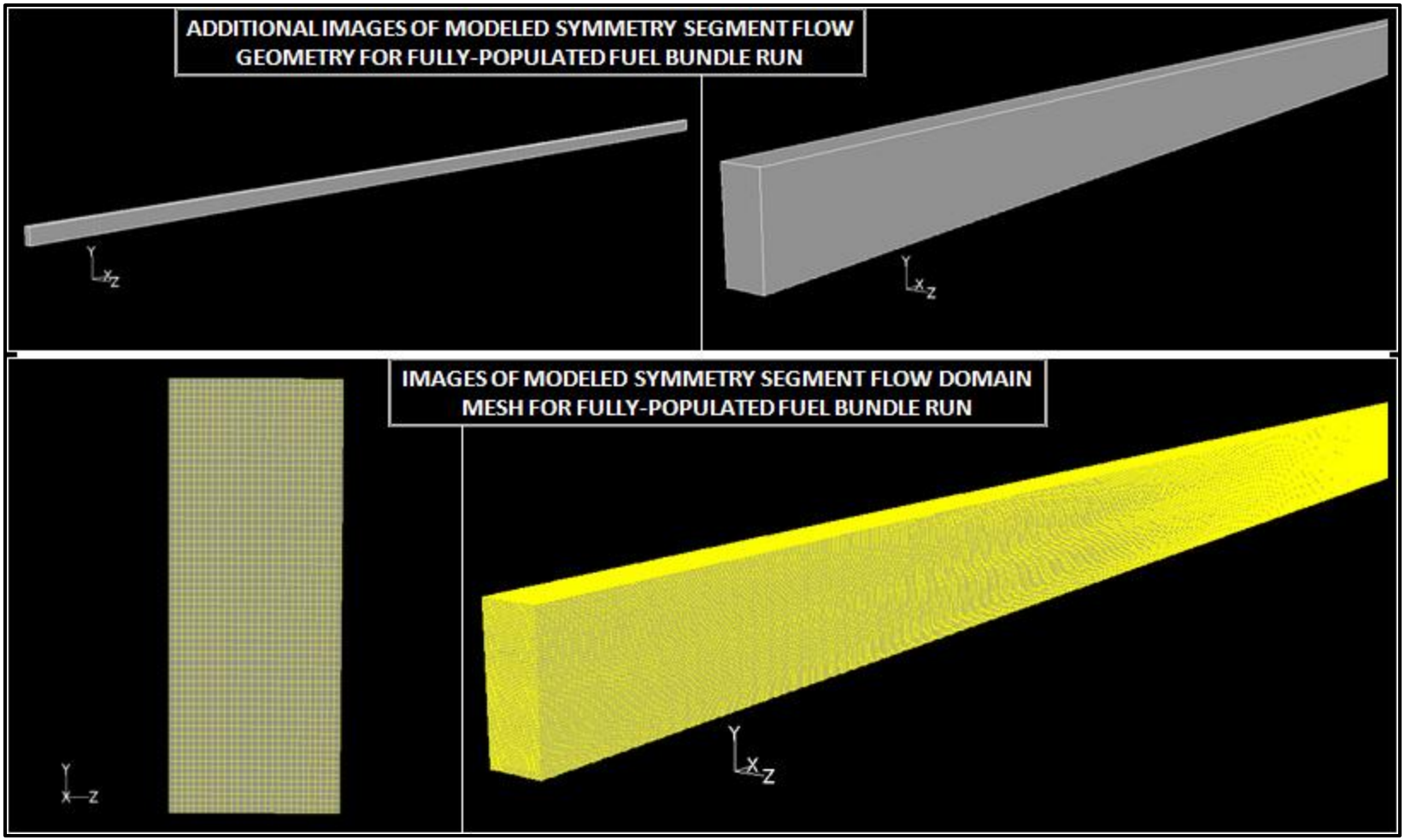


Figure 5.120:Additional images of modeled cross-sectional flow geometry and domain mesh representing a fully-populated bundle run symmetry segment for equivalent fuel assembly model.

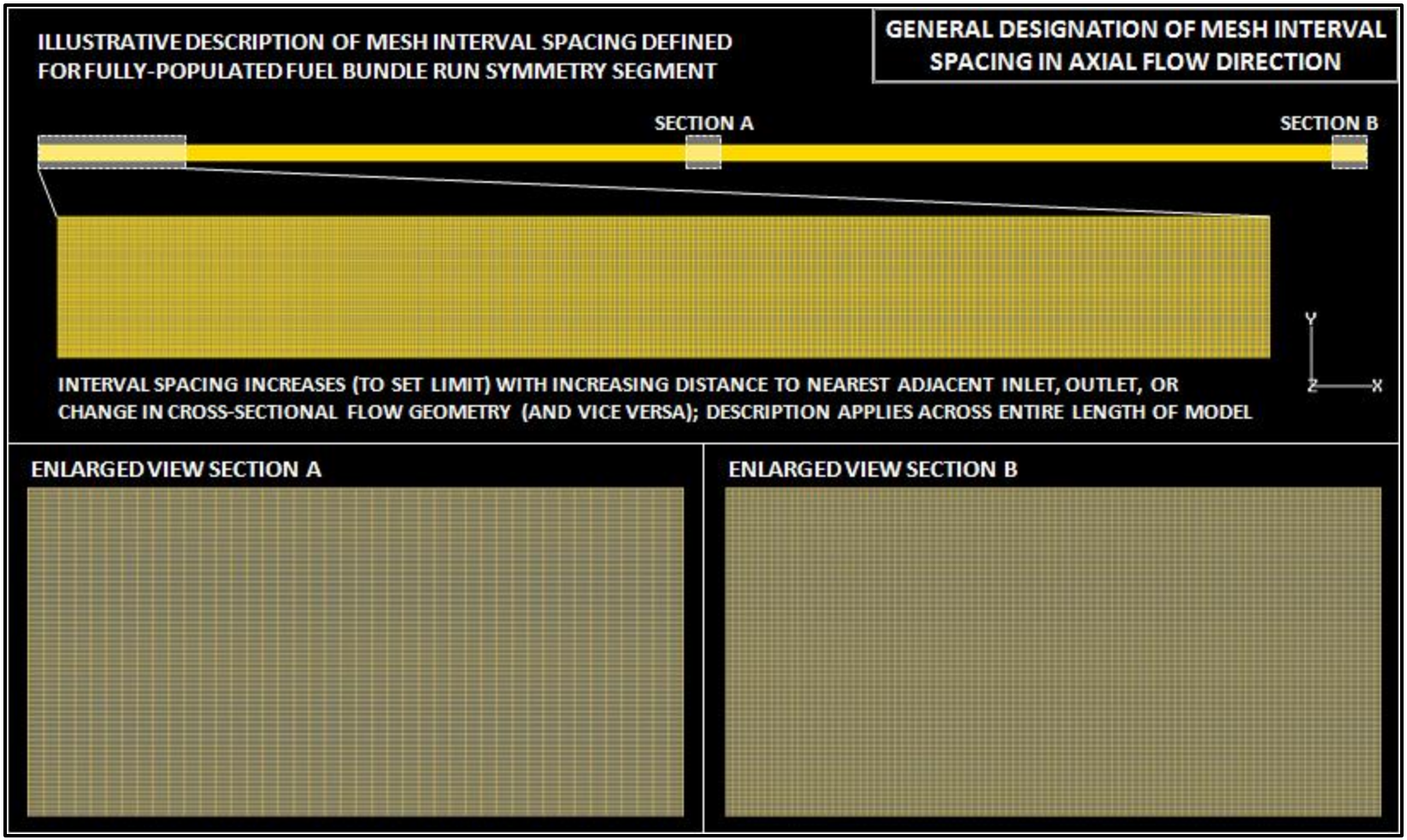


Figure 5.121:Reference example mesh of fully-populated bundle run symmetry segment illustrating the general assignment of mesh interval spacing (in axial flow direction) across length of equivalent fuel assembly model.

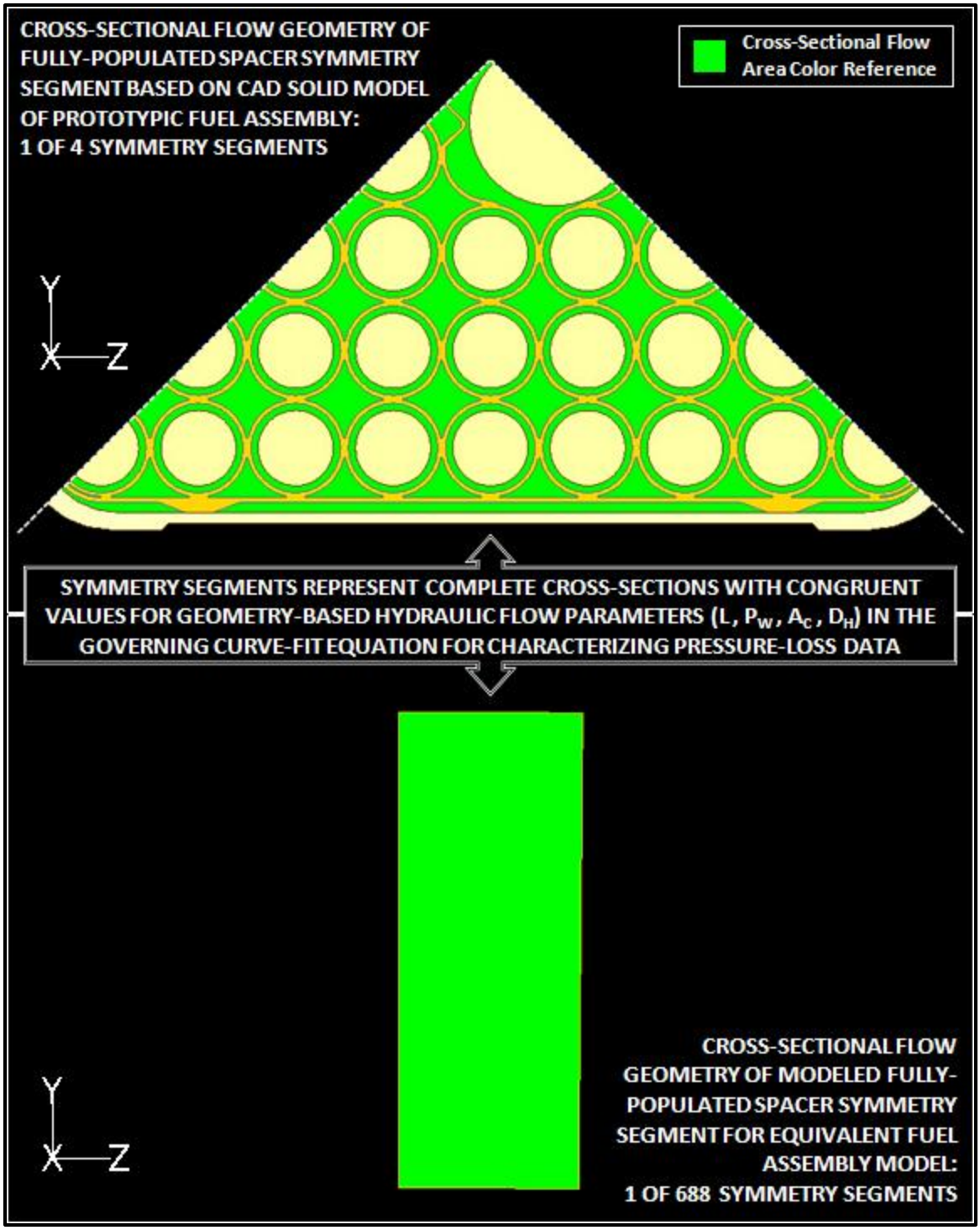


Figure 5.122:Cross-sectional flow domain comparison of fully-populated spacer symmetry segments from CAD solid and equivalent fuel assembly models.

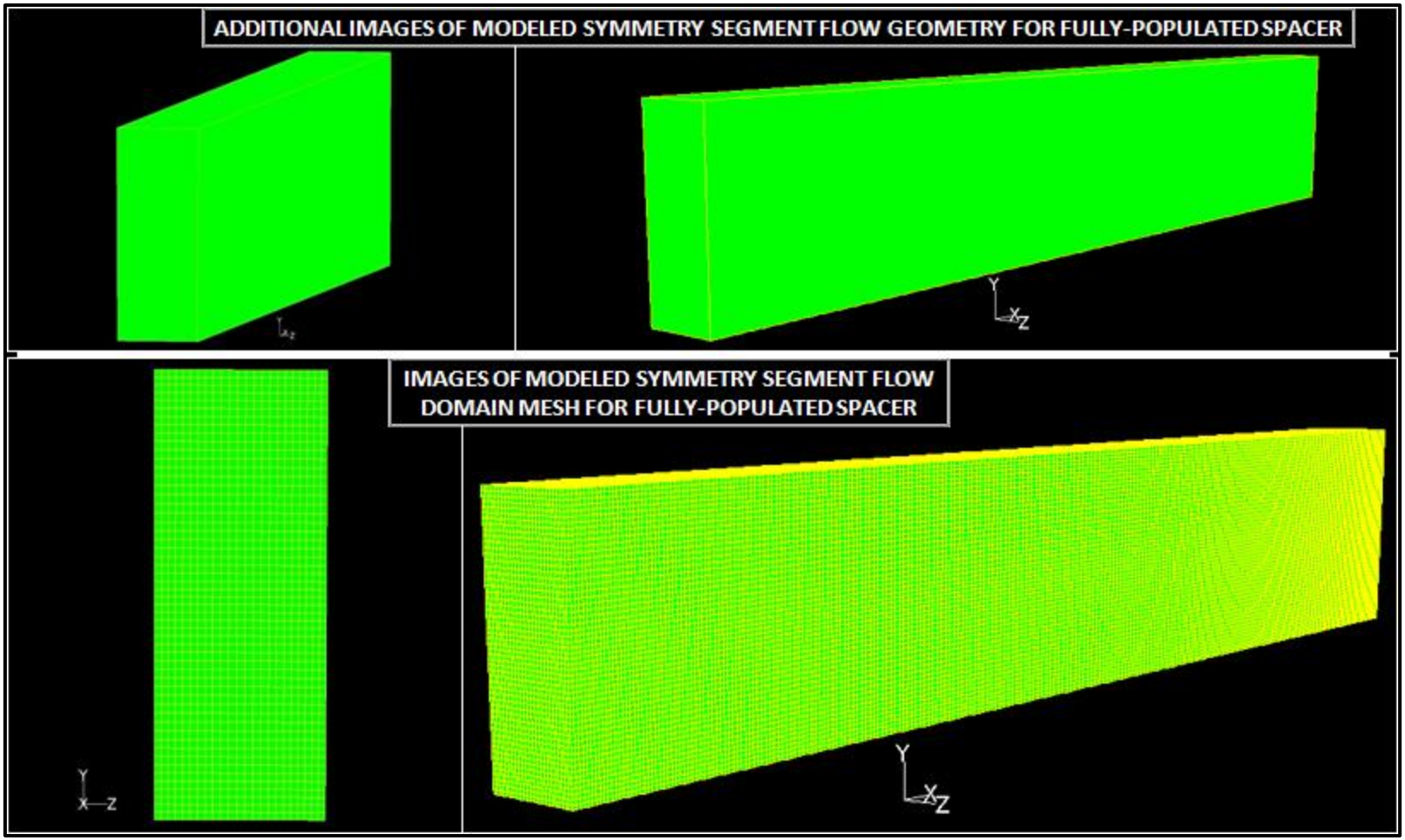


Figure 5.123: Additional images of modeled cross-sectional flow geometry and domain mesh representing a fully-populated spacer symmetry segment for equivalent fuel assembly model.

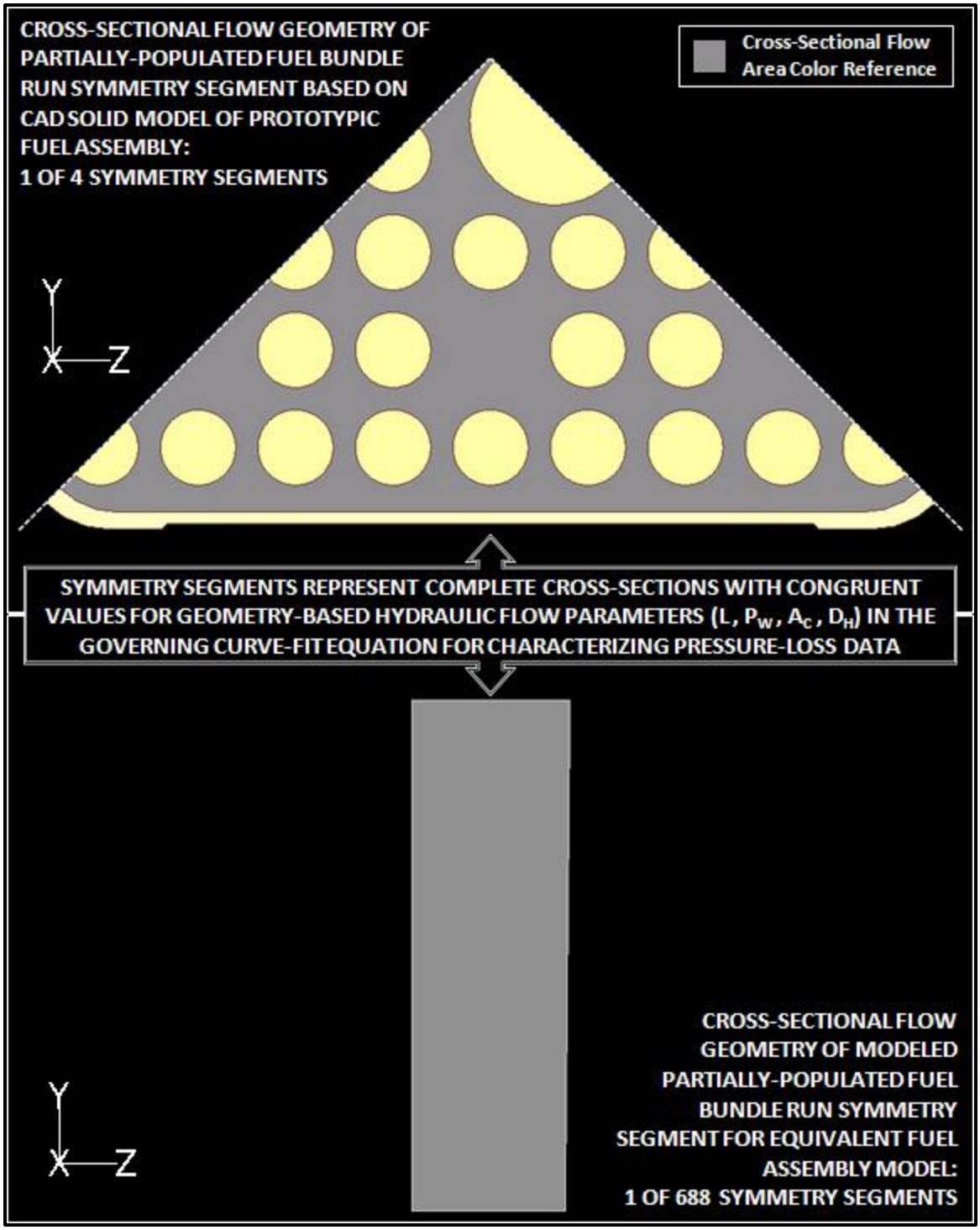


Figure 5.124:Cross-sectional flow domain comparison of partially-populated bundle run symmetry segments from CAD solid and equivalent fuel assembly models.

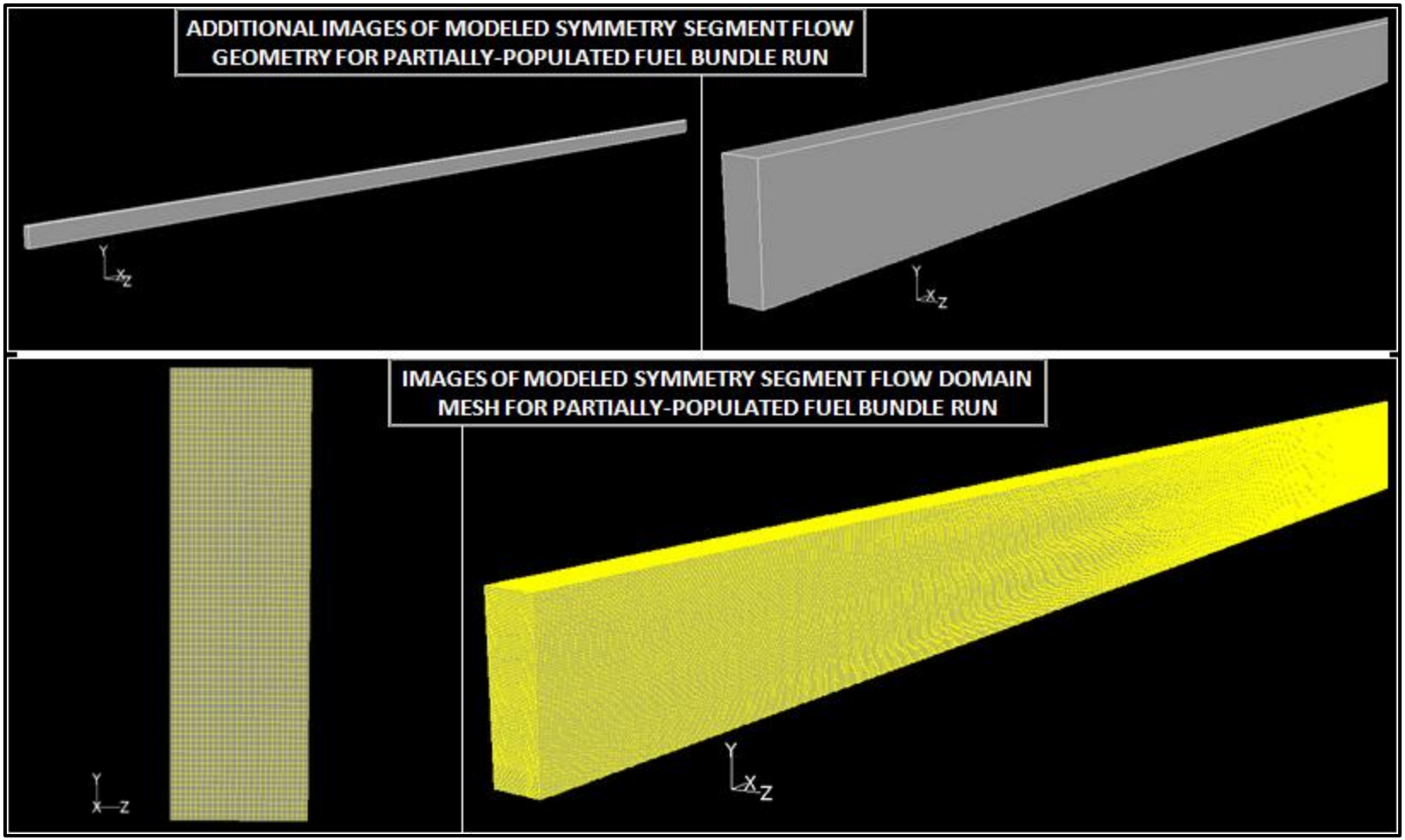


Figure 5.125:Additional images of modeled cross-sectional flow geometry and domain mesh representing a partially-populated bundle run symmetry segment for equivalent fuel assembly model.

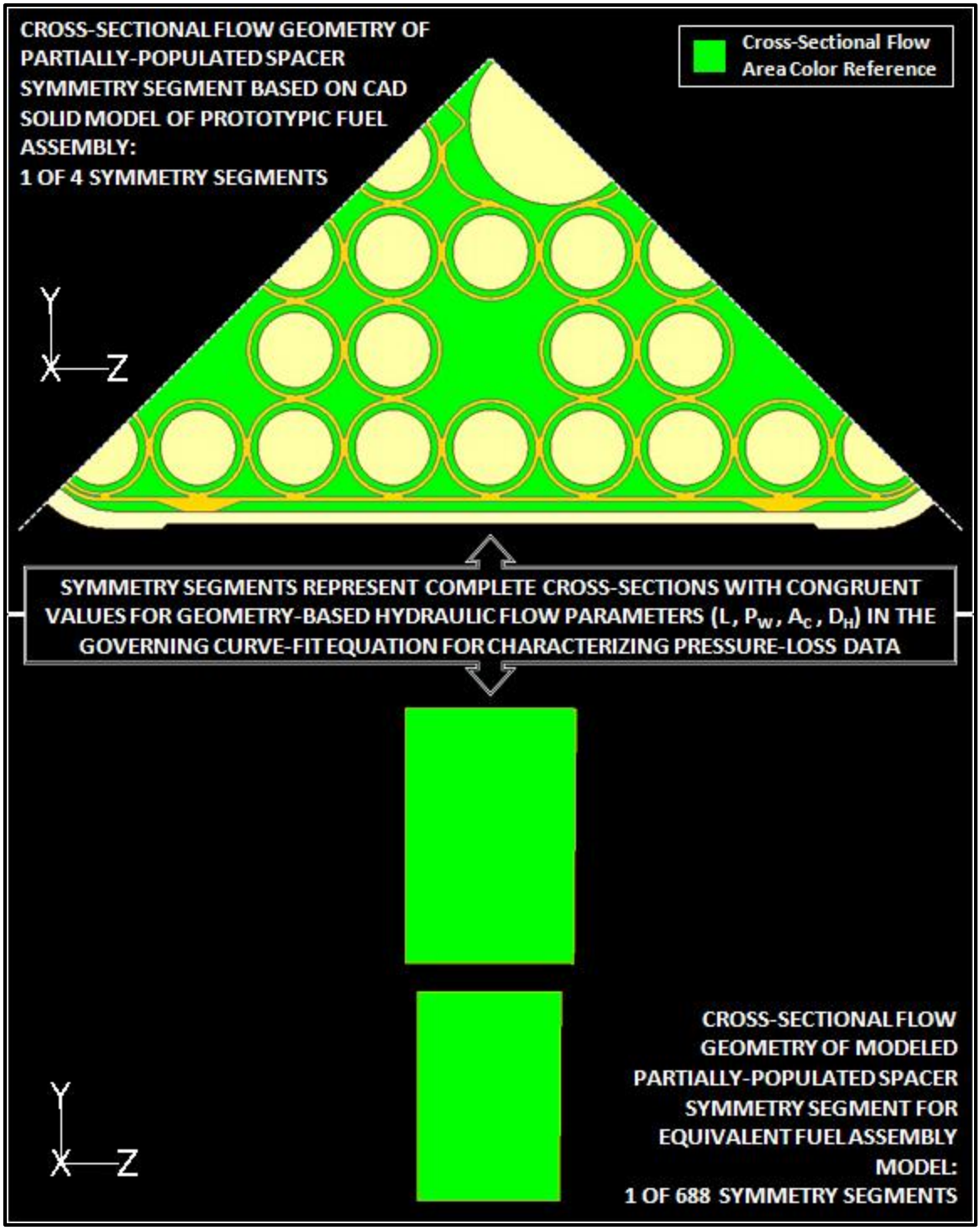


Figure 5.126:Cross-sectional flow domain comparison of partially-populated spacer symmetry segments from CAD solid and equivalent fuel assembly models.

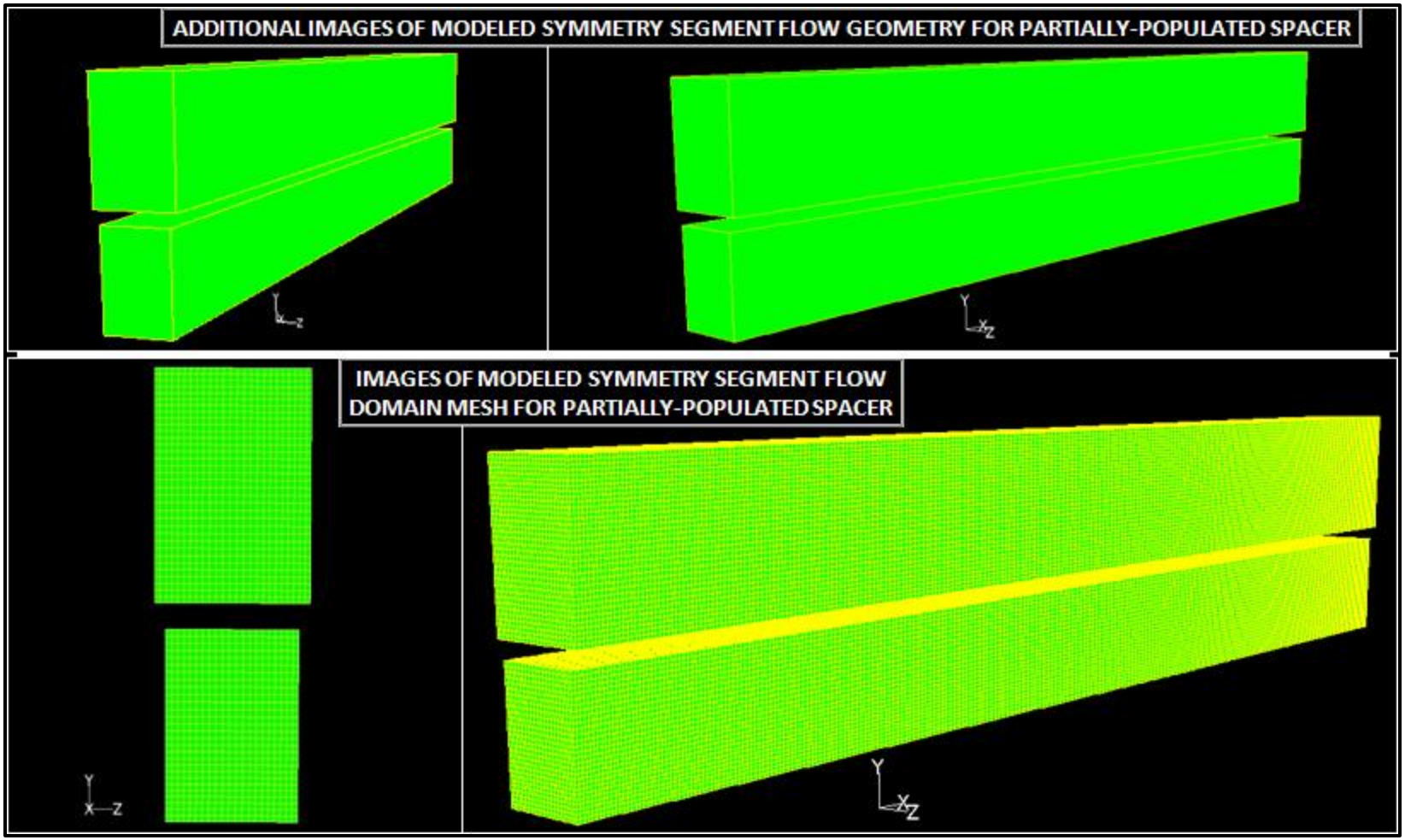


Figure 5.127:Additional images of modeled cross-sectional flow geometry and domain mesh representing a partially-populated spacer symmetry segment for equivalent fuel assembly model.

Tables 5.14 and 5.15 below contain a final summary of the initial modeling considerations and pertinent characteristic aspects addressed above throughout associated descriptions concerning the preliminary model basis for conducting the ensuing simulation trial runs and comparative evaluations.

Table 5.14: Equivalent fuel assembly model reference geometry and mesh characteristics (1 of 2).

EQUIVALENT FUEL ASSEMBLY MODEL - GEOMETRY AND MESH CONSTRUCTION DETAILS - REFERENCE VALUES AND NOTES

[Design Basis]

Each component enclosed by the volume domain comprising the equivalent fuel assembly model is constructed with base geometrical dimensions that completely satisfy the hydraulic characteristics and flow lengths defined for each such component within the developed CAD solid model representation of a GNF 9x9 BWR fuel assembly (described previously in Section 4.1); as before, spacer values are ascertained from geometry at component midplane section. Any deviations due to model simplifications are otherwise noted below, however, all modifications are minute and assumed to incur negligible effect on solution predictions for overall fuel assembly system. See accompanying figures detailing the model geometry for additional reference.

[Mesh Basis]

A high-quality mesh representation of the model flow domain is achieved by following the general principles and recommendations outlined previously in Section 4.2 for generating grid elements of high-quality metrics and sufficient cell number densities where necessitated within volume segments to adequately resolve the critical thermal-hydraulic flow features that develop across the modeled domain. See accompanying figures detailing the described mesh model for additional reference.

[Extent Basis]

Domain enclosed by the equivalent fuel assembly model represents a span length between lower and upper tie plates of the fuel assembly apparatus with an extent defined by SNL pressure ports (2-17) explicitly.

[Geometry - Components]	No. Entities ():	Length (m):	Flow Area (m^2):	Hydraulic Diameter (m):
Fully-Populated Fuel Bundle Run	5	0.473330	9.688292E-03	0.01190721
Fully-Populated Grid Spacer	5	0.030480	7.267489E-03	0.00290797
Partially-Populated Fuel Bundle Run	3	0.473330	1.046384E-02	0.01406727
Partially-Populated Grid Spacer	2	0.030480	8.251381E-03	0.00359172

Table 5.15: Equivalent fuel assembly model reference geometry and mesh characteristics (2 of 2).

[Geometry - Model]	Fully-Populated Length (m):	Partially-Populated Length (m):	Total Length (m):	No. Symmetry Planes ():
Equivalent Fuel Assembly Domain	2.519050	1.480950	4.00	344

NOTE: Although minute differences exist among fuel bundle run lengths, to simplify model construction, a single length is assigned to all fuel bundle run components (both fully- and partially-populated) based upon the average and combined total length of all such components across the fuel assembly domain. Furthermore, variations in water rod diameters across the fully-populated inlet and partially-populated outlet fuel bundle runs are negated within the modeled domain; therefore, identical representations are defined for fuel bundle run components respective to each fully- or partially-populated section of fuel assembly. Also, lengths reported for individual components in the SNL hydraulic analysis are defined according to pressure port locations and not actual component lengths as tabulated dimensions represent above for model components. As a result, apparent notable discrepancies between component lengths arise out of this representative distinction for defining the component extents, but are not indicative of actual deviations in the genuine geometry designated for fuel assembly components.

[Mesh - Components]	No. Mesh Entities ():	Volume Elements ():	Volume Mesh Scheme Types ():	No. Volume Cells ():	Max EAS ():	Max AR ():	Max SC ():
Fully-Populated Fuel Bundle Run	5	Hexahedral	Cooper	3090204	0.002526	3.625600	1.076760
Fully-Populated Grid Spacer	5	Hexahedral	Map, Cooper	306774	0.002527	2.155590	1.045190
Partially-Populated Fuel Bundle Run	3	Hexahedral	Cooper	3511300	0.015721	3.705620	1.101120
Partially-Populated Grid Spacer	2	Hexahedral	Map, Cooper	360938	0.015727	2.200860	1.075500

[Mesh - Model]	No. Individual Mesh Files Appended ():	Case Mesh [RAM] Required (GB):	No. Volume Cells ():	Max EAS ():	Max AR ():	Max SC ():
Equivalent Fuel Assembly Domain	8	≈ 35	28240666	0.015727	3.705620	1.101120

NOTE: Notations used above for mesh quality-type specifications include EquiAngle Skew (EAS), Aspect Ratio (AR), and Size Change (SC).

5.3.2 Unheated Pressure-Driven Flow Trials

Initial computational experiments focus on unheated, pressure-driven airflows through the equivalent fuel assembly domain at the same flow rates evaluated in the SNL physical experiments. Tables 5.16 and 5.17 below describe solver and simulation setup details as well as other pertinent reference parameters associated with the unheated simulation trials conducted.

Table 5.16: FLUENT CFD solver and simulation setup details for unheated, pressure-driven flow trials using the equivalent fuel assembly model (1 of 2).

UNHEATED EQUIVALENT FUEL ASSEMBLY MODEL - FLUENT CFD SOLVER AND SIMULATION SETUP DETAILS WITH REFERENCE PARAMETERS

[Solver Details]	Description:
Solver Selection	Pressure-Based
Precision	Double
Formulation	Implicit
Space	3D
Time	Steady
Gradient Option	Green-Gauss Node-Based
Viscous Model	Laminar
Pressure-Velocity Coupling	SIMPLE

[Solver Details - Continued]	Description:
Equations	Flow
Pressure Discretization Scheme	Second-Order
Momentum Discretization Scheme	Second-Order Upwind
Pressure Under-Relaxation	0.3
Density Under-Relaxation	1.0
Body Forces Under-Relaxation	1.0
Momentum Under-Relaxation	0.7

[Convergence Criteria / Monitors]	Description:
Scaled Residuals (All Equations)	1.0E-06
Mass Imbalance (% Net Domain Flux)	« 0.1
Pressure Surface Monitors (Inlet-Outlet)	Negligible Δ ≈ 100 Iterations

[Operating Conditions / Properties]	Description:
Constant Air Density (kg/m^3)	0.981415
Constant Air Viscosity (kg/m-s)	1.820975E-05
Operating Pressure (Pa)	82833.7622
Gravitational Acceleration (m/s^2)	-9.81

Table 5.17: FLUENT CFD solver and simulation setup details for unheated, pressure-driven flow trials using the equivalent fuel assembly model (2 of 2).

[Boundary Types]	Description:	[Boundary Conditions]	Description:
Model Flow Inlet	Velocity Inlet	**Velocity Inlet (See Note Below)**	Component Profiles
Model Flow Outlet	Pressure Outlet	**Pressure Outlet**	Defaults
Solid-Fluid Interface	Wall	**Wall**	Defaults
Symmetry Plane	Symmetry	**Symmetry**	Defaults

NOTE: Since the velocity profile downstream of the lower tie plate is unknown at the model inlet location, an assumption is required with regard to the velocity inlet specification that is prescribed for comparison against SNL experimental measurements. Therefore, a reasonable estimate of the actual flow downstream of the lower tie plate is assumed to resemble the partially developed velocity profile at the outlet of a half-length, fully-populated fuel bundle segment from the equivalent assembly model geometry with a uniform velocity inlet condition. The resulting outlet flow profiles from these initial half-length, fuel bundle model runs (same solver formulation) are subsequently implemented as the x-, y-, and z-component profiles for velocity inlet boundary conditions at corresponding flow rates examined using the equivalent fuel assembly model.

[Simulation Trials]	Description:
Flow Rates Examined (slpm)	200, 300, 400, 500, and 600

Simulation trials for unheated flow losses across a representative assembly domain using the equivalent fuel assembly model are performed for airflow rates in the range of 200 to 600 slpm. Figure 5.128 below illustrates the comparison between FLUENT CFD simulation results and those reported by SNL from experimental measurements obtained in consideration of unheated airflow rates spanning this specified range of values. The resulting pressure loss data compare within 10% over the entire range of fuel bundle airflow velocities, which constitutes a sufficient level of agreement between the computational and SNL experimental flow loss responses for this fuel assembly structure. However, a discussion of the observed discrepancies among compared solution values is included in Section 5.3.4 below for additional reference.

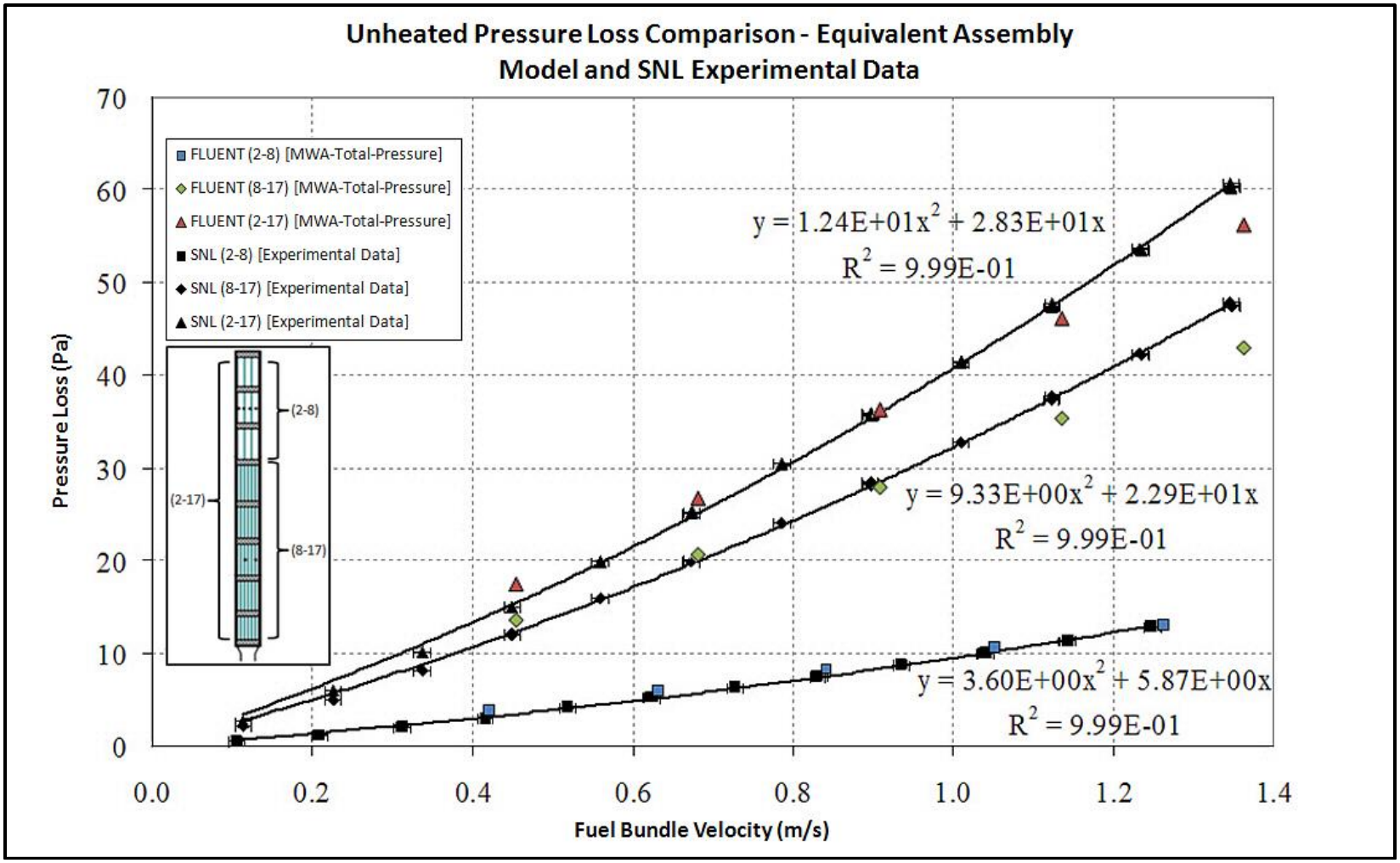


Figure 5.128:Comparison of unheated pressure losses from equivalent fuel assembly model and SNL experimental trials.

Furthermore, subsequent figures and tables provided below include an accompaniment of graphical comparisons as well as numerical quantities for principal flow-field parameters acquired from the unheated simulation trial solutions, which are central to the hydraulic analysis and curve-fit derivation of requisite loss coefficient values. As such, Figures 5.129 and 5.130 depict results for cumulative pressure loss per assembly component and grouped segment data for component pressure loss per segment airflow velocity, respectively. Again, these data are employed in determining the hydraulic loss coefficient values of each fuel assembly segment modeled. Numerical quantities for the data plotted in Figures 5.129 and 5.130 are presented in Tables 5.18 and 5.19 for reference. Foremost, in conclusion, Table 5.20 reveals the final curve-fit parameters and loss coefficient values obtained from these unheated simulation trial solutions through utilization of the TableCurve 2D analysis software.

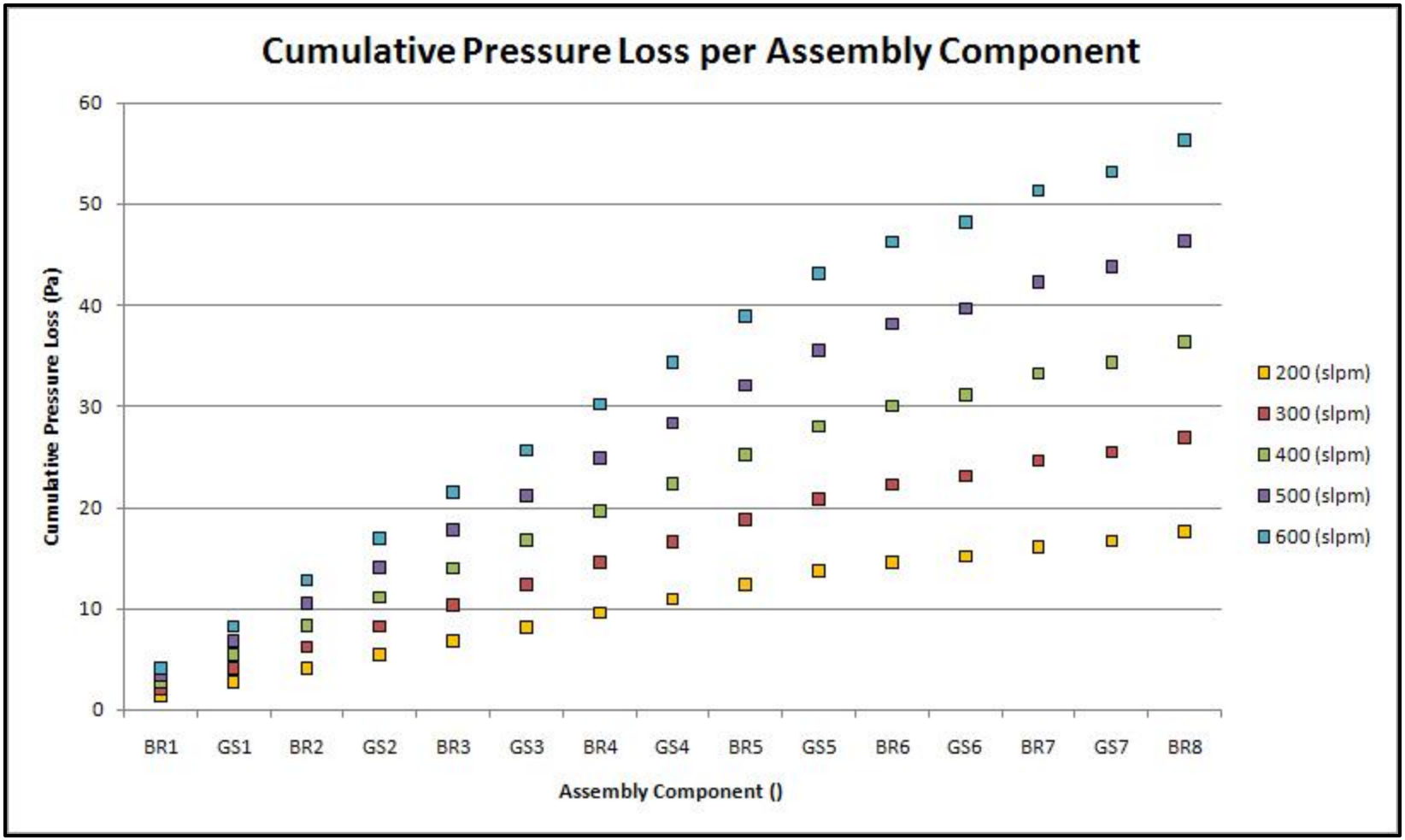


Figure 5.129:Cumulative pressure loss (MWA) per assembly component for unheated, pressure-driven flow trials using equivalent fuel assembly model.

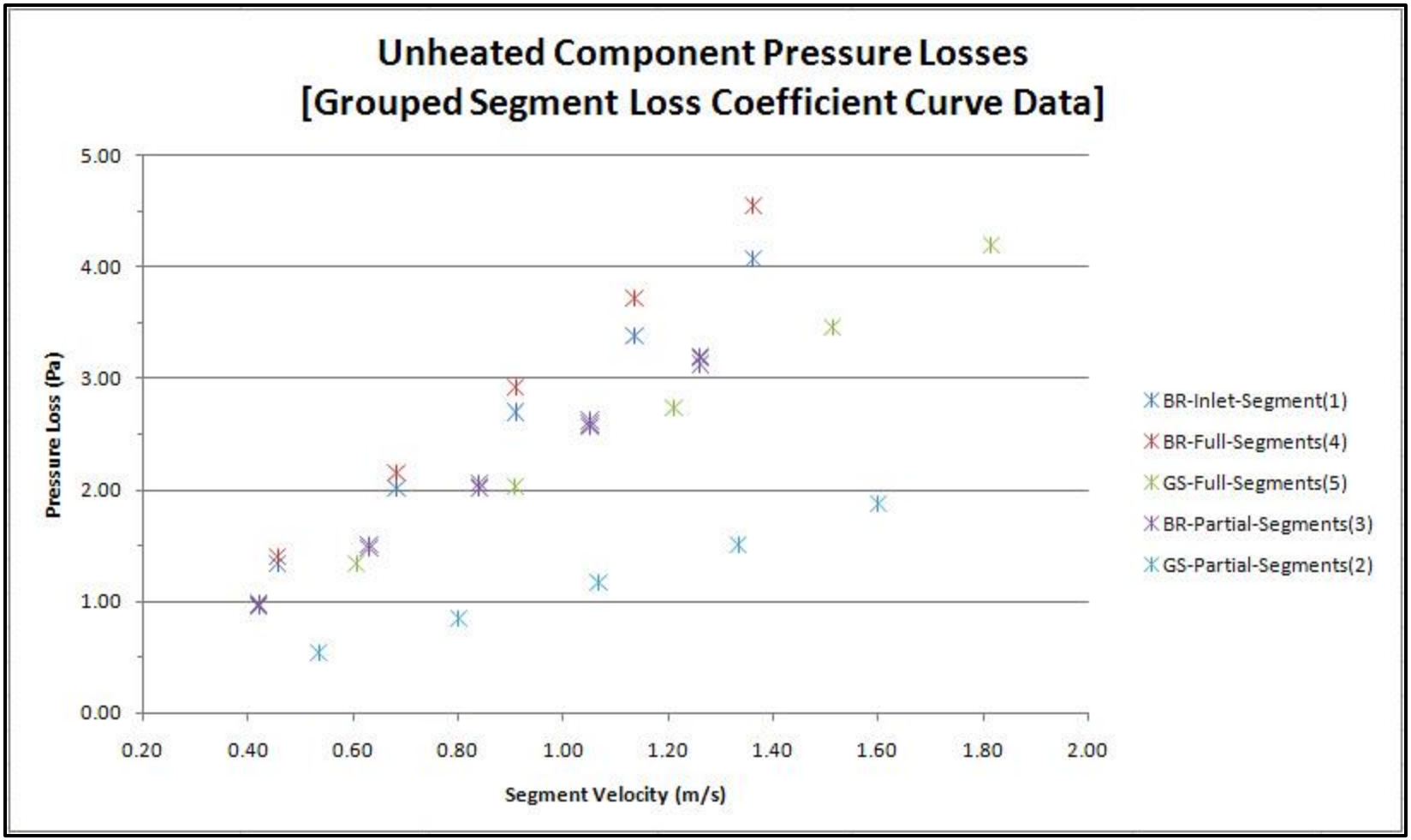


Figure 5.130:Unheated component pressure loss (MWA) data for segment velocities used to determine the hydraulic loss coefficient values of grouped segments.

Table 5.18: Pertinent FLUENT CFD simulation results from unheated, pressure-driven flow trials using the equivalent fuel assembly model (1 of 2).

UNHEATED EQUIVALENT FUEL ASSEMBLY MODEL - PERTINENT SIMULATION TRIAL RESULTS						
[Velocity (m/s)]		200 slpm	300 slpm	400 slpm	500 slpm	600 slpm
Fully-Populated Fuel Bundle Run		0.45385897	0.68078845	0.90771794	1.13464742	1.36157691
Fully-Populated Grid Spacer		0.60503955	0.90755932	1.21007910	1.51259887	1.81511864
Partially-Populated Fuel Bundle Run		0.42022037	0.63033056	0.84044074	1.05055093	1.26066111
Partially-Populated Grid Spacer		0.53289483	0.79934224	1.06578966	1.33223707	1.59868448
[MWA Total Pressure Loss (Pa)]		200 slpm	300 slpm	400 slpm	500 slpm	600 slpm
Inlet Fuel Bundle Run 1:	BR1	1.3412730	2.0164470	2.6942870	3.3756830	4.0607800
Grid Spacer 1:	GS1	1.3397600	2.0304810	2.7353970	3.4534920	4.1848330
Fuel Bundle Run 2:	BR2	1.4016400	2.1450310	2.9156840	3.7134620	4.5372320
Grid Spacer 2:	GS2	1.3397620	2.0304810	2.7353990	3.4534880	4.1848330
Fuel Bundle Run 3:	BR3	1.4016400	2.1450310	2.9156840	3.7134630	4.5372320
Grid Spacer 3:	GS3	1.3397608	2.0304820	2.7353970	3.4534890	4.1848320
Fuel Bundle Run 4:	BR4	1.4016399	2.1450300	2.9156860	3.7134610	4.5372330
Grid Spacer 4:	GS4	1.3397618	2.0304820	2.7353980	3.4534890	4.1848310
Fuel Bundle Run 5:	BR5	1.4016399	2.1450299	2.9156840	3.7134630	4.5372340
Grid Spacer 5:	GS5	1.3398705	2.0306573	2.7356411	3.4537950	4.1851960
Fuel Bundle Run 6:	BR6	0.9553728	1.4749208	2.0212054	2.5930523	3.1887400
Grid Spacer 6:	GS6	0.5391137	0.8426771	1.1686097	1.5145616	1.8781800
Fuel Bundle Run 7:	BR7	0.9765347	1.5075428	2.0569760	2.6211024	3.1981054
Grid Spacer 7:	GS7	0.5391137	0.8426776	1.1686106	1.5145628	1.8781819
Fuel Bundle Run 8:	BR8	0.9588899	1.4790250	2.0164868	2.5675907	3.1305884

Table 5.19: Pertinent FLUENT CFD simulation results from unheated, pressure-driven flow trials using the equivalent fuel assembly model (2 of 2).

[Cumulative Losses (Pa)]		200 slpm	300 slpm	400 slpm	500 slpm	600 slpm
Assembly Section (8-17):	BR1 → GS5	13.646748	20.749152	28.034257	35.497285	43.134236
Assembly Section (2-8):	BR6 → BR8	3.969025	6.146843	8.431889	10.810870	13.273796
Entire Assembly (2-17):	BR1 → BR8	17.615773	26.895996	36.466146	46.308155	56.408032

Table 5.20: Hydraulic loss coefficient results from unheated, pressure-driven airflow trials for grouped (flow path) segments using the equivalent fuel assembly model.

EQUIVALENT FUEL ASSEMBLY MODEL - TABLECURVE 2D CURVE FIT DATA - UNHEATED HYDRAULIC LOSS COEFFICIENTS					
[BR-Inlet-Segment]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.941034	0.030252	1.0000	96.7569	0.061650
[BR-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.969000	0.267182	1.0000	97.6770	0.544484
[GS-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.169390	0.075111	1.0000	66.1042	0.153067
[BR-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.200103	0.253063	0.9993	101.0240	0.515710
[GS-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	0.934320	0.151016	1.0000	43.4322	0.307751

5.3.3 Heated Buoyancy-Driven Flow Trials

The aforementioned comparison of results for unheated, pressure-driven airflows from CFD simulations and SNL experiments clearly demonstrates a sufficient level of congruency in the pressure loss responses to substantiate the existence of a numerically significant relationship between flow-field developments across an actual BWR fuel assembly structure and those predicted by simulation using the equivalent fuel assembly model. As a result, the capability of drawing meaningful characterizations for describing typical fuel assembly responses through comparisons of varying accident conditions and external governing parameters is truly a plausible realization through application of this computationally efficient, simplified fuel assembly model construct. Intending to exploit this potential capability and computational benefit, ensuing simulation trials focus upon heated, buoyancy-driven airflows through the equivalent fuel assembly domain for comparative analysis with foregoing, unheated flow-field developments and loss coefficient values.

Specifically, these runs simulate the decay heat-driven natural convection airflows that possibly exist/evolve within a SFP and contained fuel assembly structures following a complete loss of water coolant inventory. Tables 5.21 through 5.24 below provide a summary description of solver and simulation setup details as well as other pertinent reference parameters associated with the heated simulation trials conducted.

Table 5.21: FLUENT CFD solver and simulation setup details for heated, buoyancy-driven flow trials using the equivalent fuel assembly model (1 of 4).

HEATED EQUIVALENT FUEL ASSEMBLY MODEL - FLUENT CFD SOLVER AND SIMULATION SETUP WITH REFERENCE PARAMETERS

[Solver Details]	Description
Solver Selection	Pressure-Based
Precision	Double
Formulation	Implicit
Space	3D
Time	Steady
Gradient Option	Green-Gauss Node-Based
Viscous Model	Laminar
Pressure-Velocity Coupling	SIMPLE
Equations	Flow, Energy

[Solver Details: Continued]	Description
Pressure Discretization Scheme	Body-Force-Weighted
Momentum Discretization Scheme	Second-Order Upwind
Energy Discretization Scheme	Second-Order Upwind
Pressure Under-Relaxation	0.7
Density Under-Relaxation	0.99
Body Forces Under-Relaxation	0.99
Momentum Under-Relaxation	0.4
Energy Under-Relaxation	0.95

[Convergence Criteria - Monitors]	Description
Scaled Residual (Continuity)	1.0E-05
Scaled Residuals (All Velocity)	1.0E-08
Scaled Residual (Energy)	1.0E-08
Mass Imbalance (% Net Domain Flux)	« 0.1
MFR Surface Monitors (Inlet-Outlet)	Negligible Δ ≈ 4000 Iterations
Max Temp Volume Monitors (All Zones)	Negligible Δ ≈ 4000 Iterations

[Solution Limits]	Description
Minimum Temperature (K)	294.0
Maximum Temperature (K)	3000.0
Minimum Absolute Pressure (Pa)	1.0
Maximum Absolute Pressure (Pa)	5.0E+10

Table 5.22: FLUENT CFD solver and simulation setup details for heated, buoyancy-driven flow trials using the equivalent fuel assembly model (2 of 4).

NOTE: Since the initialized solution for the flow domain coincides with a uniform air mass equal to the ambient inlet temperature and property set, buoyancy source terms are initially negligible and gradually build during computation. This likely results in very small initial continuity and momentum residuals that lead to small scale factors or large starting scaled residuals. Therefore, to aid numerical convergence hindered by the gradual build-up of source terms from initializing the solution at ambient uniform conditions, after a few initial iterations, reasonable conservative estimates for temperature and velocity are patched into each bundle run and spacer cell zone segment across the modeled domain. However, as a result, a degree of uncertainty is introduced into the utilization of these equation residuals as strong metrics for judging convergence or even just simply setting appropriate criteria limits. In conclusion, the above scaled residual criteria for solved equations are set conservatively stringent and heavy emphasis is placed on monitoring integrated quantities such as mass flow rate as well as maximum cell zone temperatures (see analysis figures) throughout the entire domain for negligible variances (*i.e.* 0.1% or less change over 4000 iterations) in these flow-field values to more-accurately gauge convergence across simulation runs.

[Operating Conditions]	Description	
Operating Density (kg/m^3)	0.98141468	(Set to outer-assembly ambient environment)
Operating Pressure (Pa)	82833.7622	(Set to outer-assembly ambient environment)
Gravitational Acceleration (m/s^2)	- 9.81	

[Fluid - Solid Material Properties]	Description	
Air Density (kg/m^3)	Incompressible Ideal Gas Law	(FLUENT-defined option for density relationship)
Air Specific Heat Capacity (J/kg-K)	Piecewise-Polynomial, $f(T)$	(Values defined by temp-dependent function; see note below)
Air Thermal Conductivity (W/m-K)	Piecewise-Polynomial, $f(T)$	(Values defined by temp-dependent function; see note below)
Air Dynamic Viscosity (kg/m-s)	Piecewise-Polynomial, $f(T)$	(Values defined by temp-dependent function; see note below)
INCONEL-625 Density (kg/m^3)	8440.0	(Constant value)
INCONEL-625 Sp. Heat Capacity (J/kg-K)	Piecewise-Polynomial, $f(T)$	(Values defined by temp-dependent function; see note below)
INCONEL-625 Thm. Conductivity (W/m-K)	Piecewise-Polynomial, $f(T)$	(Values defined by temp-dependent function; see note below)

Table 5.23: FLUENT CFD solver and simulation setup details for heated, buoyancy-driven flow trials using the equivalent fuel assembly model (3 of 4).

NOTE: Aside from density, fluid (air) and solid (INCONEL-625) material properties are defined by piecewise-polynomial functions of temperature. These temperature-dependent functions are derived by fitting tabulated data sets of property values from Incropera and DeWitt (2002) and Special Metals Corporation (2006), respectively.

[Specified Boundary Types]	Description
Model Flow Inlet	Pressure Inlet
Model Flow Outlet	Pressure Outlet
Bounding Solid-Fluid Interface	Wall
Symmetry Plane	Symmetry

[Boundary Setup: Pressure Inlet]	Description
Gauge Total Pressure (Pa)	0.0
Supersonic-Initial Gauge Pressure (Pa)	0.0
Direction Specification Method	Normal-To-Boundary
Total Temperature (K)	294.4034

[Boundary Setup: Wall]	Description
Wall Motion	Stationary Wall
Shear Condition	No Slip
Solid Material	INCONEL-625
Thermal Conditions	Constant Zone Heat Flux
Heat Flux (W/m^2): Walls Normal To Flow	0.0
Heat Flux (W/m^2): Walls Remaining	Zone Dependent (See Note)

[Boundary Setup: Pressure Outlet]	Description
Gauge Pressure (Pa)	0.0
Backflow Direction Spec. Method	Normal-To-Boundary
Backflow Total Temperature (K)	294.4034

NOTE: Heat flux values for wall zones along axial flow direction are dependent upon net heat input rates specified for the fuel assembly domain as well as the corresponding actual and representative segment geometries defined by the CAD solid and equivalent fuel assembly models, respectively. These boundary condition input values at associated wall zones are described below for each simulation trial and net heat input rate examined.

Table 5.24: FLUENT CFD solver and simulation setup details for heated, buoyancy-driven flow trials using the equivalent fuel assembly model (4 of 4).

[Simulation Trials]	Trial 1	Trial 2	Trial 3	Trial 4	*Trial 5*
Net Heat Input Rate (W)	300.0	600.0	900.0	1200.0	1500.0
Fully-Pop. Bundle Heat Flux (W/m^2)	24.00515	48.01031	72.01546	96.02062	120.02577
Fully-Pop. Spacer Heat Flux (W/m^2)	7.815332	15.63066	23.44600	31.26133	39.076660
Partially-Pop. Bundle Heat Flux (W/m^2)	23.41923	46.83846	70.25769	93.67692	117.09616
Partially-Pop. Spacer Heat Flux (W/m^2)	7.582802	15.16560	22.74841	30.33121	37.914012

(Maximum temperature solution limit reached during computation; no results obtained for simulation trial)

NOTE: Heat flux inputs above pertain to wall zones extending in the axial flow direction for each of the four different flow segment geometries comprising the entire domain of the equivalent fuel assembly model. Heat input rates for individual flow path segments are based on a fraction of the total length of all fuel rods in a flow segment to the total length of all fuel rods in the entire fuel assembly, which is multiplied by the net heat input rate for a certain simulation trial; total number and length of fuel rods in a particular flow path segment correspond with the aforementioned CAD solid model representation and defined flow segment length from the equivalent fuel assembly model geometry, respectively. The heat input rate for each flow path segment is divided by its associated wall zone surface area defined within the equivalent fuel assembly model in order to determine the final heat flux input value expressed above for each net heat input rate examined.

In contrast to the hydraulic analysis of loss coefficients for unheated assembly airflows, the material properties expressed in the defined curve-fit equation no longer constitute constant values across resulting solution data sets for heated, buoyancy-driven airflows. Therefore, consideration for the temperature dependence of these air properties in buoyancy-driven systems necessitates an alternative formulation of the original curve-fit equation used to resolve the loss coefficient values. The derivation is facilitated via a straightforward approach from Equation (3.9) by initially arranging the variable (inconstant) terms into a discrete subset within each product group representing the major and minor pressure losses as follows:

$$\Delta P_{total} = \frac{S_{LAM} L}{2 D_h^2} (\mu V) + \frac{\sum k}{2} (\rho V^2) \tag{5.33}$$

where, the variable term subsets enclosed in parentheses are central to the ensuing development effort. As such, the primary task is simply to recast the above stated curve-fit expression into a second-order polynomial form by replacing all variable terms on the right-hand side of Equation (5.33) with a single parameter definition.

Accordingly, a constraint imposed by the implementation of a single parameter embodiment for these variable terms requires that corresponding subsets encompass a congruent multiple of common factors on the right-hand side of the equality. Hence, in order to appropriately address this constraint, Equation (5.33) is multiplied through by a select ratio of variable terms denoted as (ρ/μ^2) and simplified to obtain a suitable expression of the form

$$\frac{2\Delta P_{total}\rho}{\mu^2} = \frac{S_{LAM} L}{D_h^2} \left(\frac{\rho V}{\mu}\right) + \sum k \left(\frac{\rho^2 V^2}{\mu^2}\right) \tag{5.34}$$

Furthermore, composing subsets of inconstant terms in this precise arrangement infers a second-order polynomial materialization from Equation (5.34), which is conceivable by substituting a single notional parameter for the common factor multiple in both sets of parentheses, yielding

$$\frac{2\Delta P_{total}\rho}{\mu^2} = \frac{S_{LAM}L}{D_h^2}(C) + \sum k\,(C^2) \tag{5.35}$$

where, the parameter (C) is clearly defined as

$$C = \frac{\rho V}{\mu} \tag{5.36}$$

The physical meaning of terms identified in Equations (5.35) and (5.36) are largely unchanged from the general explication provided in Sections 3.3 and 5.2 concerning the curve-fit expression formulated for unheated assembly airflows. However, due to the unique handling and default reporting of pressure quantities within the FLUENT CFD software for buoyant flows, several terms such as the pressure loss term (ΔP_{total}) take on a slightly different representation in the reported flow-field quantities that warrant brief clarification. In particular, the default pressure field and pressure inputs within FLUENT account for hydrostatic head by redefining the pressure in terms of a modified pressure (P') as follows (FLUENT Inc., 2006):

$$P' = P - \rho_o g x \tag{5.37}$$

or, likewise

$$\frac{\partial P'}{\partial x} = \frac{\partial P}{\partial x} - \rho_o g \tag{5.38}$$

This default designation in FLUENT allows for the inclusion of hydrostatic head within the body force term, $[(\rho - \rho_o)g]$, and exclusion from the pressure calculation under constant density conditions.

Thus, pressure inputs in FLUENT do not include hydrostatic pressure differences and reports of pressure quantities (P') do not exhibit any influence of the hydrostatic pressure (FLUENT Inc., 2006). As a result, the pressure loss quantity (ΔP_{total}) in the curve-fit equation for unheated flows, which represents the irreversible loss in mechanical energy, is simply related to the upstream and downstream reported pressures (denoted prime terms) across an assembly segment by

$$P'_{down} = P'_{up} - \Delta P_{total} \tag{5.39}$$

or, solving for the pressure loss term directly, yields

$$\Delta P_{total} = P'_{up} - P'_{down} \tag{5.40}$$

On the other hand, for heated flows involving density variations, the solution flow field inherently includes the additional influence of the gravitational body force in the reported pressure quantities, which appears as a nonzero source term $[(\rho - \rho_o)g]$ in the momentum equations. This influence is explicitly pertinent in the current model setup

with varying internal flow-field density (ρ) and constant operating density (ρ_o) set equal to the ambient column of air outside the fuel assembly structure. Therefore, although the redefinition of pressure and corresponding stipulations are still applicable as conveyed for unheated (uniform density) flow conditions, the upstream and downstream reported pressures across an assembly segment relate differently to the pressure loss quantity (ΔP_{total}) in buoyancy-driven airflows as follows

$$P'_{down} = P'_{up} + (\rho_o - \rho_m) g L - \Delta P_{total} \tag{5.41}$$

Comparing Equations (5.39) and (5.41) reveals the additional influence imparted between reported pressure quantities taken at two measurement locations from a resolved flow-field solution within the FLUENT CFD software under heated, variable-density domain conditions.

Simply stated, the reported downstream pressure difference with an upstream reference location and reported pressure in FLUENT is further characterized by the ascribed momentum source due to the gravitational body-force term in addition to incurred flow losses along the segment for buoyancy-driven fluid mediums. This connection is originally exemplified in the form of a mathematical relationship by Equation (5.41) above, where (L) and (ρ_m) are defined as the segment length and weighted average air density throughout the segment flow volume, respectively. Again, distinguishing the source contribution of gravitational force acting on density variations from the reported pressure difference is fundamentally imperative since the total head loss is physically interpreted as a loss (or irreversible conversion) in mechanical energy per unit mass between upstream and downstream segment extents.

As such, total pressure loss across an assembly segment in heated airflow trials is ascertained from the resolved flow-field solution within FLUENT based on reported quantities for upstream and downstream pressures at segment axial extents as well as the weighted average density of air encompassed by the segment volume. These flow-field values are implemented into Equation (5.41), which is rewritten in a more suitable arrangement by solving directly for the pressure loss term (ΔP_{total}) to become

$$\Delta P_{total} = \left(P'_{up} - P'_{down}\right) + (\rho_o - \rho_m) g L \tag{5.42}$$

Thus, the total pressure loss quantity is accurately achieved as requisite in Equation (5.35) for the loss coefficient curve-fit analysis of heated assembly airflows.

A final point of noteworthy discussion is the characterization of other remaining thermophysical property and velocity terms describing the fluid airflow defined in Equations (5.35) and (5.36) above, which vary throughout the flow volume of a defined assembly segment. As outlined in Equation (5.42) for segment airflow density, variable terms across an assembly segment are represented as weighted average quantities over the spanned flow volume in Equations (5.35) and (5.36), which again are a sole function of average representative temperature and mass flow rate (constant) across the segment volume as reported within FLUENT for a resolved flow-field solution. Hence, for reasons of clarification, the variable terms based on weighted average quantities in Equations (5.35) and (5.36) are re-expressed using the same letter 'm' subscript notation as employed in Equation (5.42) above. Incorporating this parameter notation into Equations (5.35) and (5.36), these statements are respectively rewritten as

$$\frac{2\Delta P_{total}\rho_m}{\mu_m^2} = \frac{S_{LAM}L}{D_h^2}(C) + \sum k\,(C^2) \tag{5.43}$$

and

$$C = \frac{\rho_m V_m}{\mu_m} \tag{5.44}$$

where, the total pressure loss (ΔP_{total}) in Equation (5.43) is acquired according to Equation (5.42) as previously discussed.

In conclusion, Equation (5.43) is recast in a final, more-generalized form that is congruent with the manner used to present Equation (3.10) and corresponding results from unheated computational and experimental airflow trials; which promotes greater consistency across exhibited heated and unheated solution sets for comparative analysis. This is accomplished by replacing combined terms on the left-hand side and constants from each group of terms on the right-hand side of Equation (5.43) with a single parameter definition that yields the following simplified equation form

$$N = a_1(C) + a_2(C^2) \tag{5.45}$$

where, the parameter (N) is defined as

$$N = \frac{2\Delta P_{total}\rho_m}{\mu_m^2} \tag{5.46}$$

and, constant parameters (a_1) and (a_2) are given by

$$a_1 = \frac{S_{LAM} L}{D_h^2} \tag{5.47}$$

$$a_2 = \sum k \tag{5.48}$$

Therefore, flow-field solution data from buoyancy-driven airflows in heated assembly trials are presented in the form of curve-fit results to Equation (5.45) for determining the (a_1) and (a_2) constants, which permit explicit calculation of loss coefficient values by appropriately solving Equations (5.47) and (5.48) as follows

$$S_{LAM} = \frac{a_1 D_h^2}{L} \tag{5.49}$$

$$\sum k = a_2 \tag{5.50}$$

The detailed explanations and equations outlined in the above discussion provide sufficient reference as well as understanding of the methodology and characterizations imposed in reporting numerical results from heated fuel assembly trials as included throughout the ensuing summary of this computational research effort.

Figures 5.131 through 5.135 below depict the onset and successful convergence of simulation trials conducted at specified net (fuel assembly) heat input rates. The maximum number of iterations required to reach suitable convergence across these case trials exceeded 250 thousand for the highest net heat input rate evaluated.

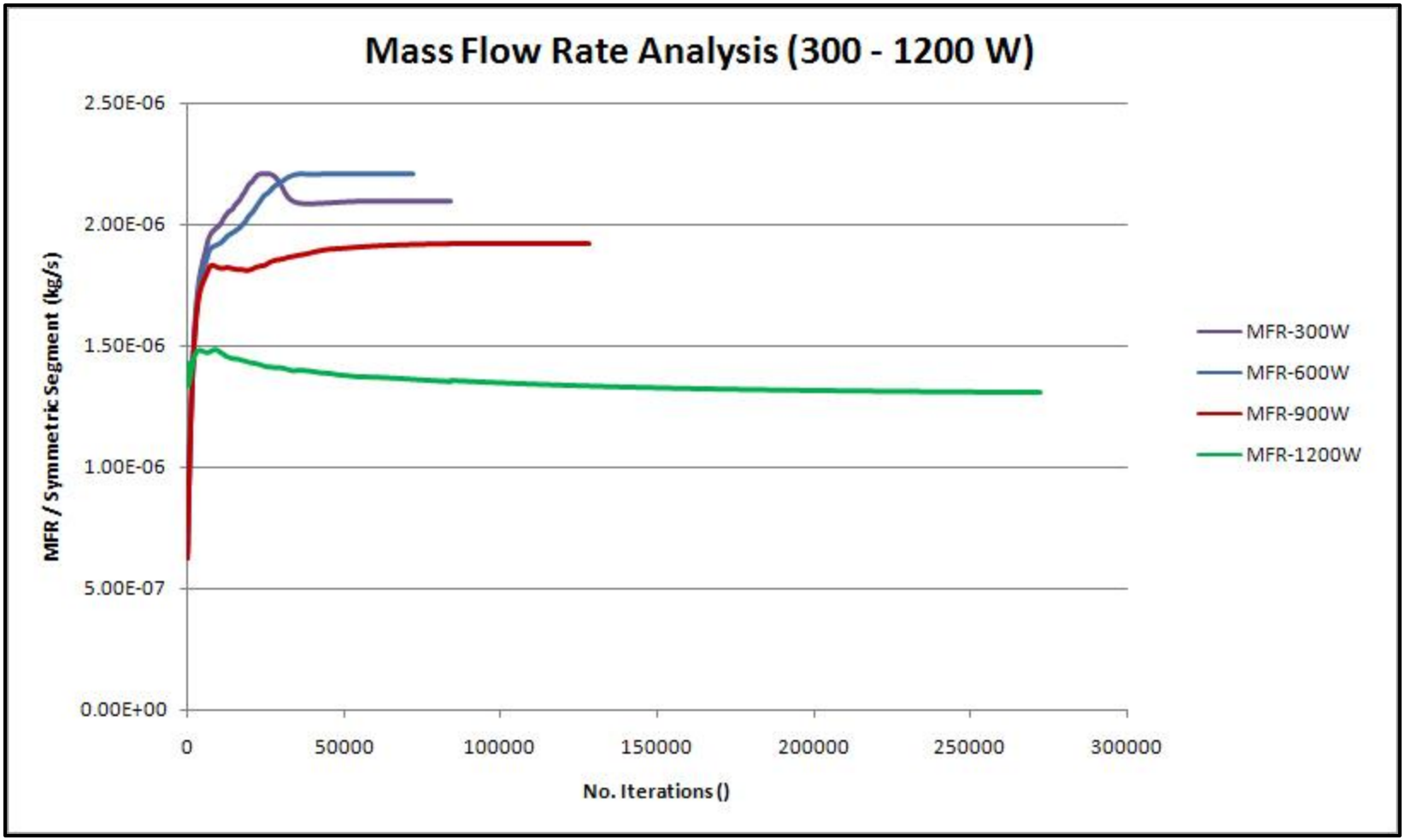


Figure 5.131:Convergence monitor analysis of the mass flow rate per symmetric segment division in equivalent fuel assembly model for heated, buoyancy-driven airflow trials.

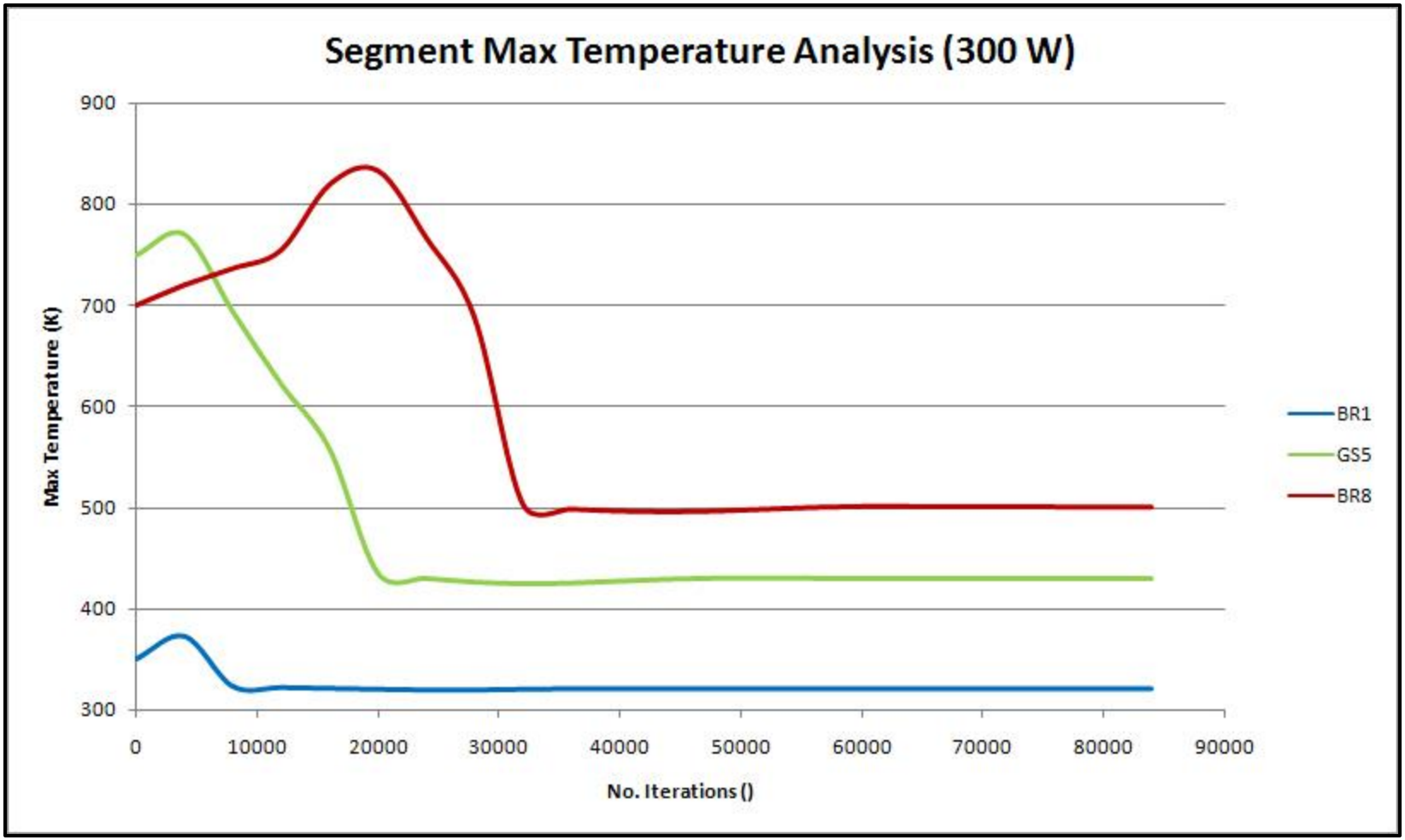


Figure 5.132:Convergence monitor analysis of segment maximum temperatures in equivalent fuel assembly model for the heated, buoyancy-driven airflow trial with net heat input rate of 300 watts.

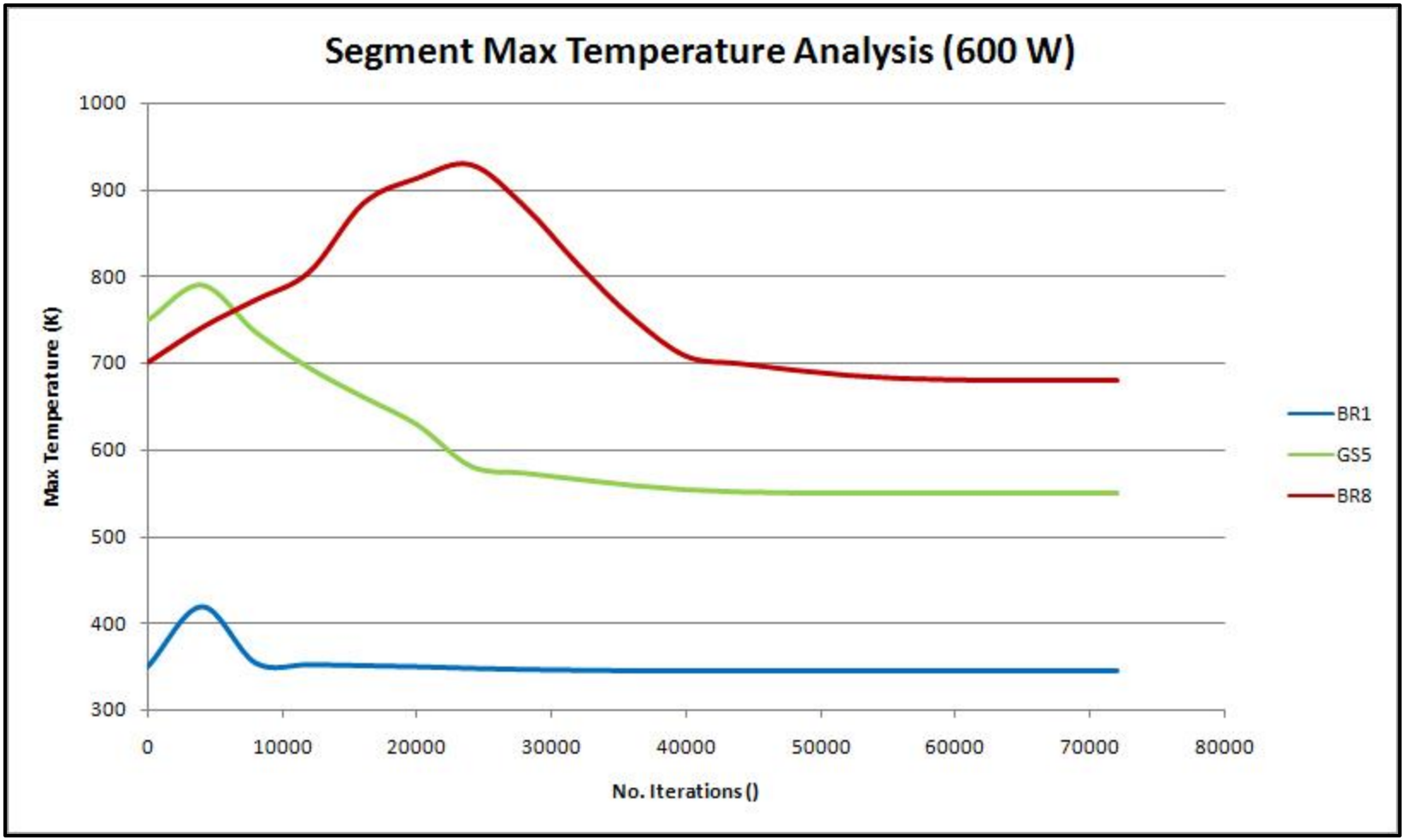


Figure 5.133:Convergence monitor analysis of segment maximum temperatures in equivalent fuel assembly model for the heated, buoyancy-driven airflow trial with net heat input rate of 600 watts.

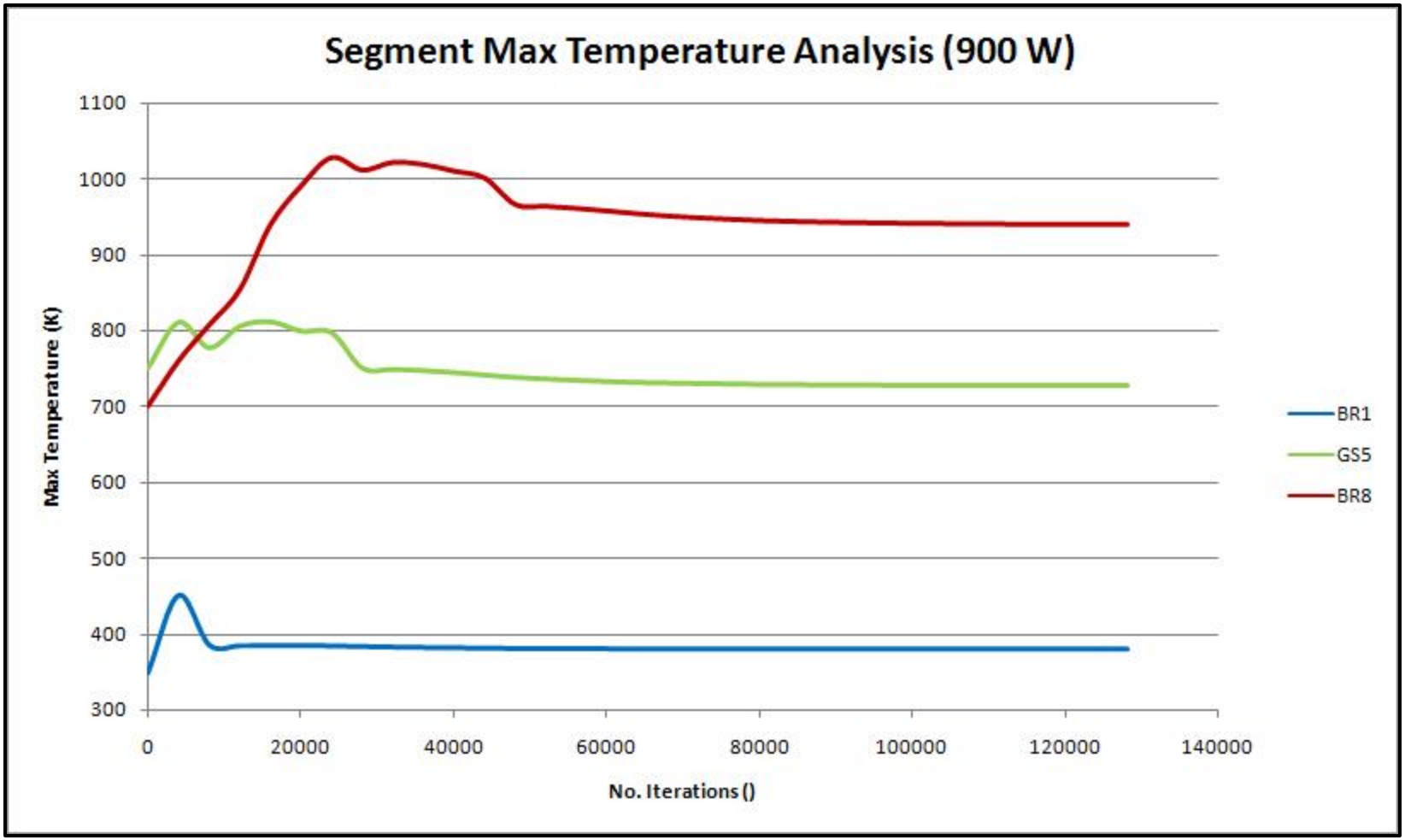


Figure 5.134:Convergence monitor analysis of segment maximum temperatures in equivalent fuel assembly model for the heated, buoyancy-driven airflow trial with net heat input rate of 900 watts.

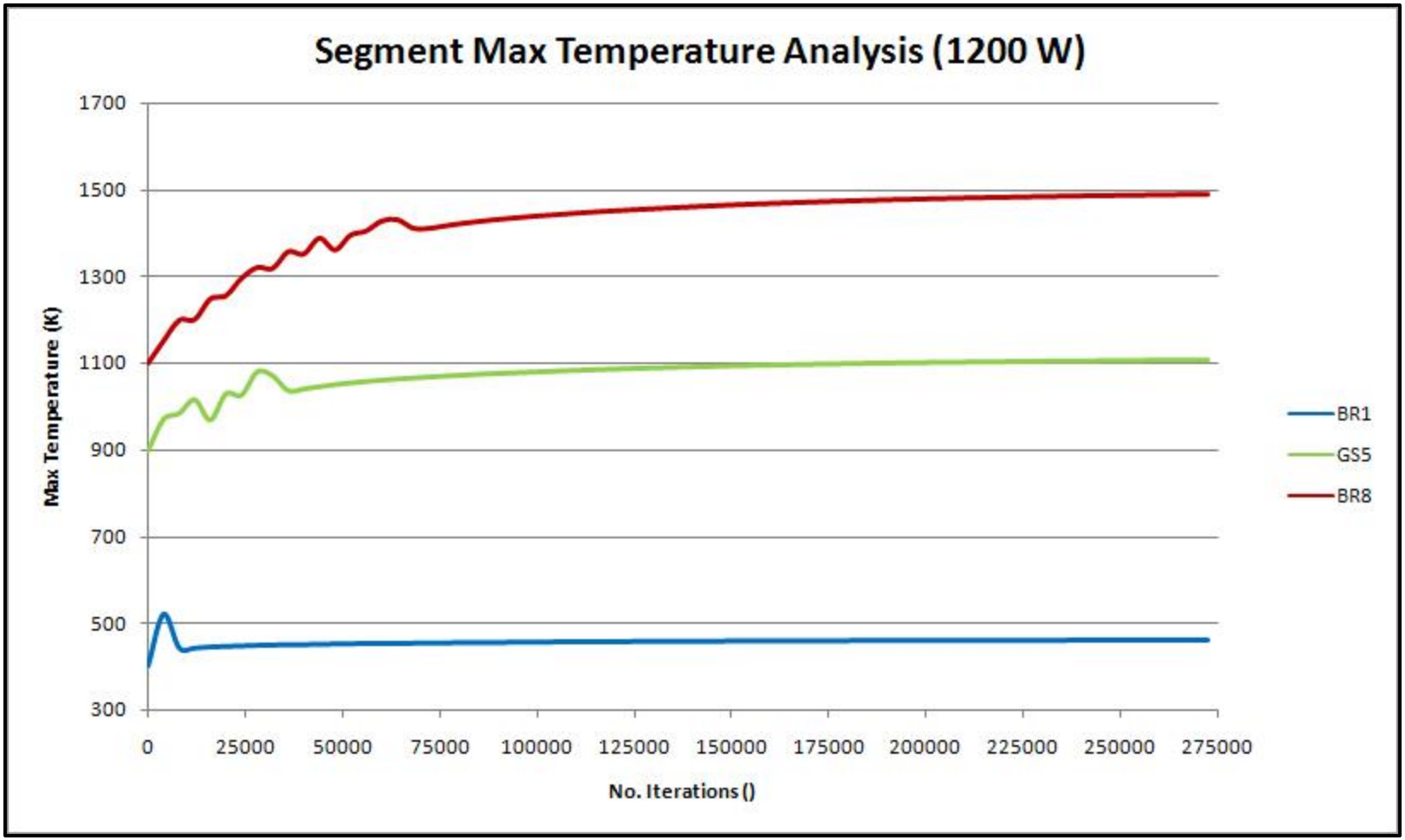


Figure 5.135:Convergence monitor analysis of segment maximum temperatures in equivalent fuel assembly model for the heated, buoyancy-driven airflow trial with net heat input rate of 1200 watts.

The following figures and tables provided below subsequently include an accompaniment of graphical comparisons as well as numerical quantities for principal flow-field parameters postprocessed from the solution data of heated FLUENT simulation trials, which are central to the hydraulic analysis and curve-fit derivation of associated loss coefficient values. As such, Figures 5.136 through 5.142 depict resultant data for key parameters requisite in discerning the dependent and independent varying terms of the curve-fit equation, the values of which are graphically exhibited in succeeding Figure 5.143 according to corresponding grouped segments. Moreover, the grouped segment data sets of Figure 5.143 are explicitly employed in determining the hydraulic loss coefficient values of each fuel assembly segment modeled.

Numerical quantities for the data plotted in Figures 5.136 through 5.143 are presented in Tables 5.25 through 5.29 for reference. Foremost, in conclusion, Table 5.30 reveals the final curve-fit parameters and loss coefficient values obtained from these heated simulation trial solutions through utilization of the TableCurve 2D analysis software.

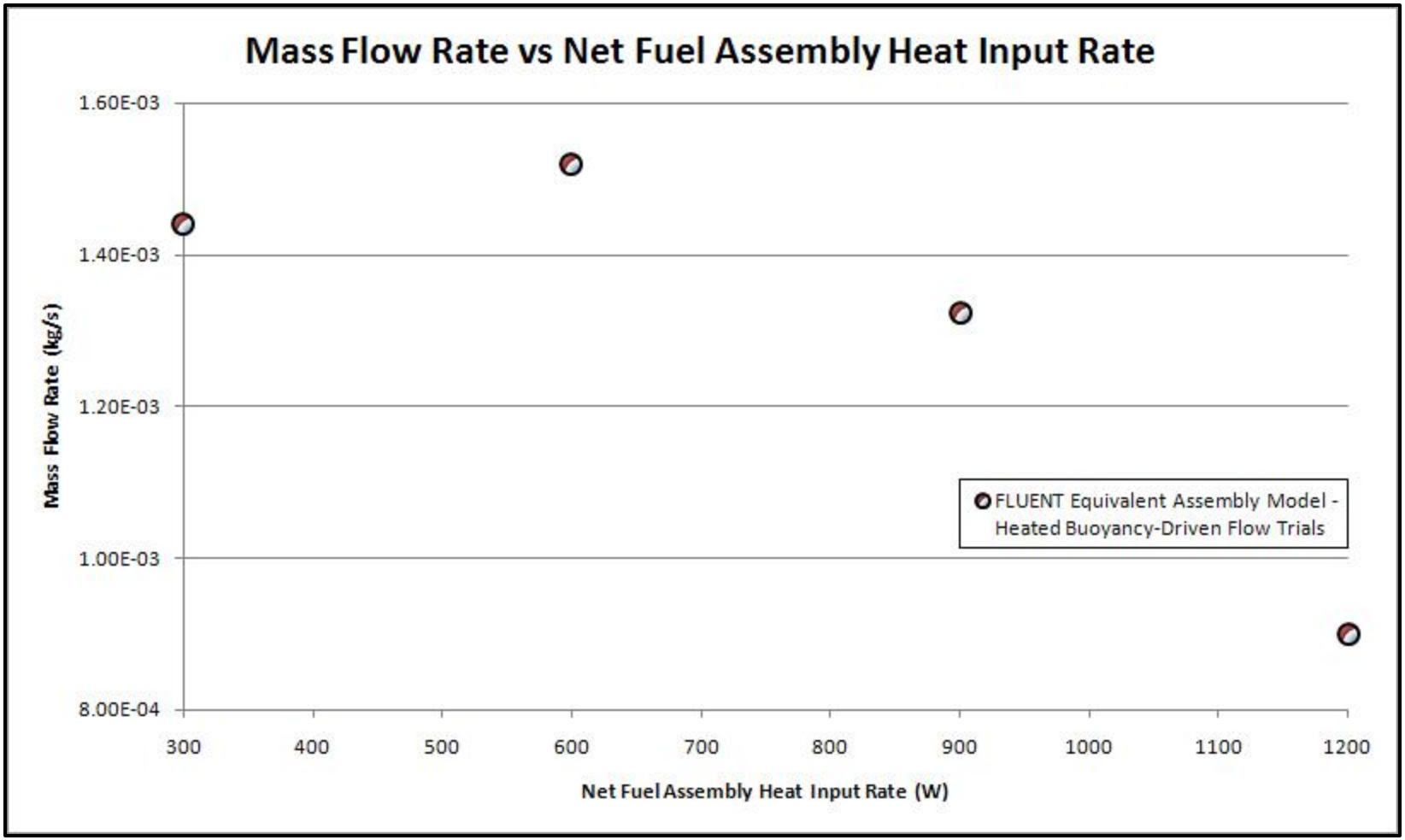


Figure 5.136:Mass flow rate versus net heat input rate for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

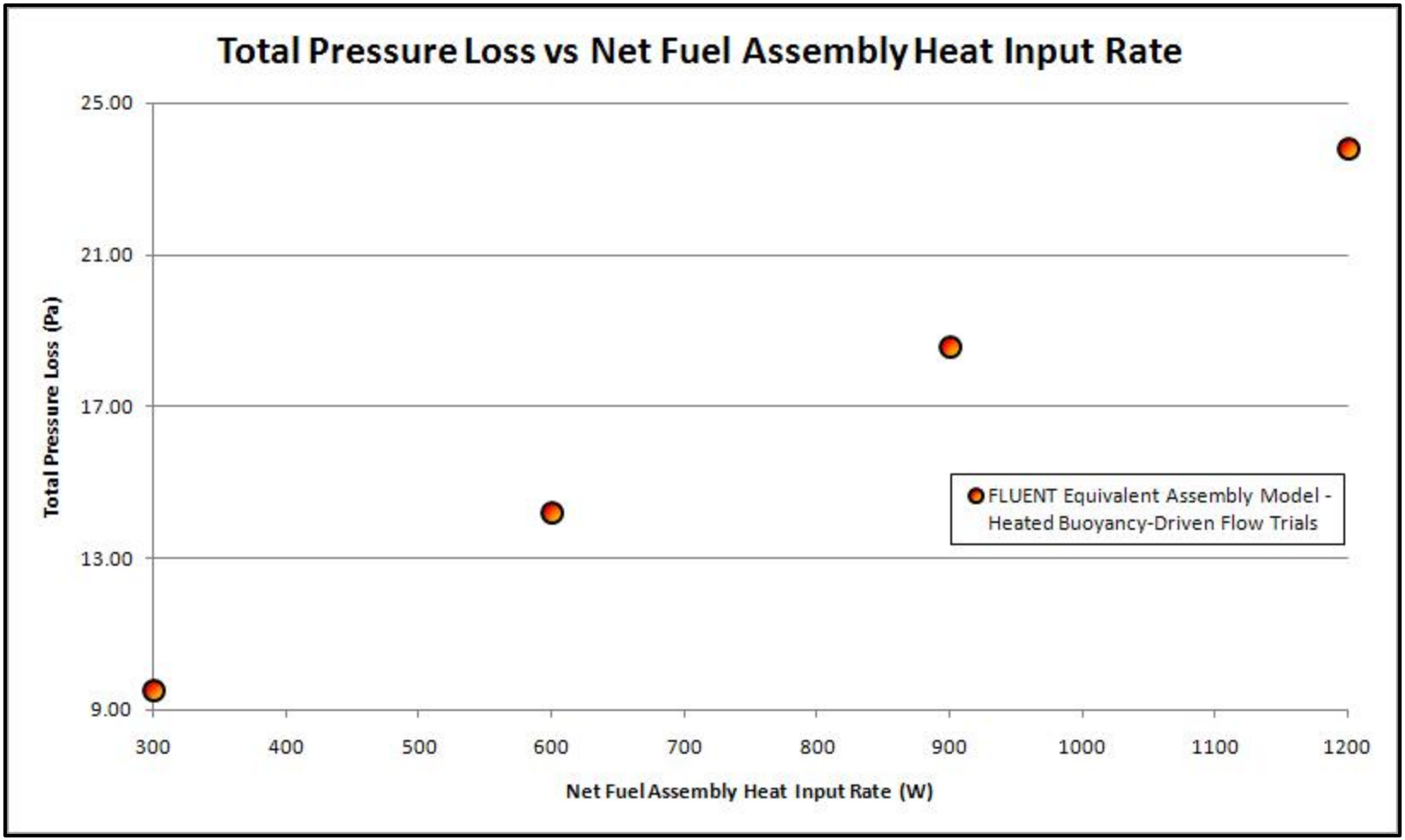


Figure 5.137:Total pressure loss (MWA) versus net heat input rate for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

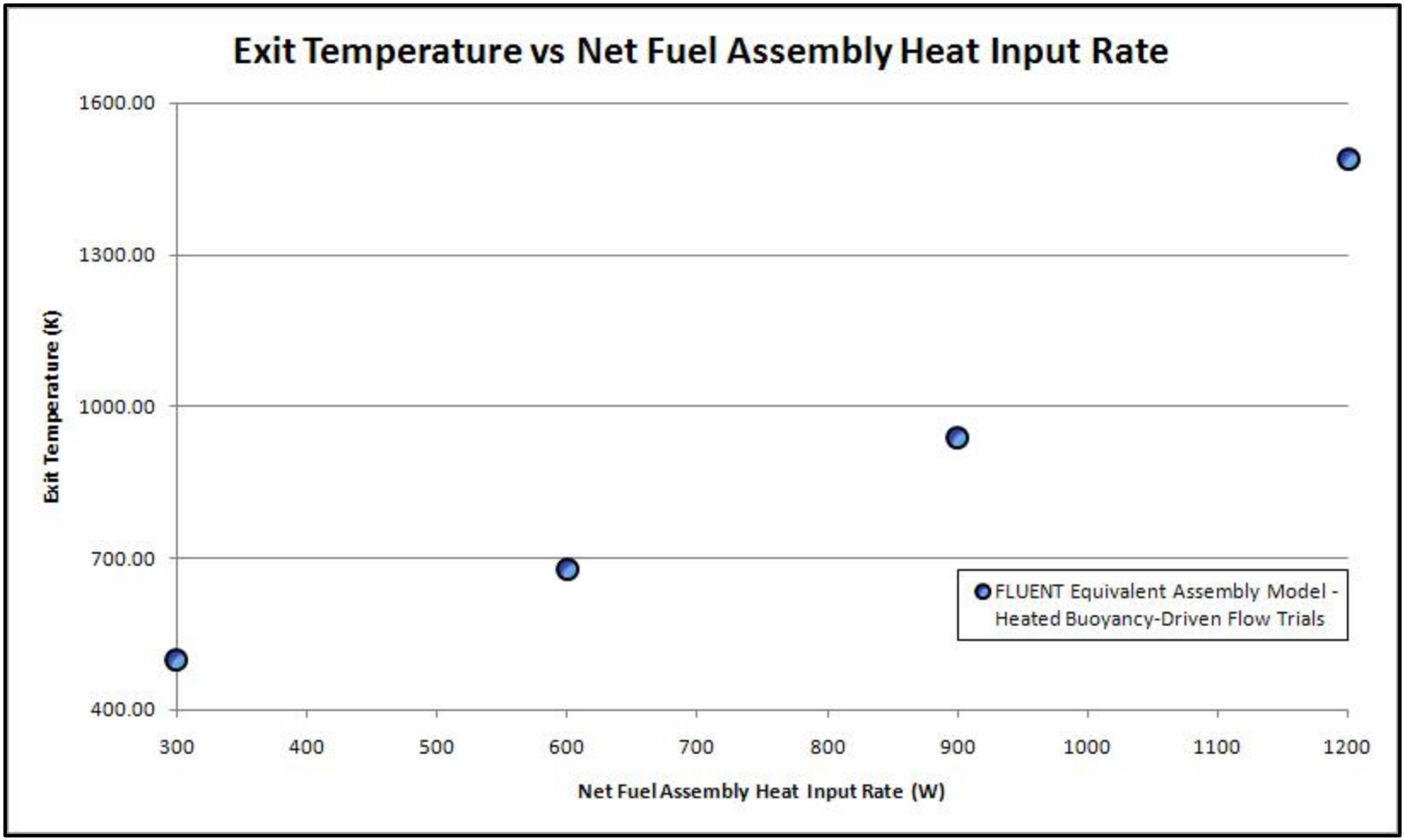


Figure 5.138:Exit temperature versus net heat input rate for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

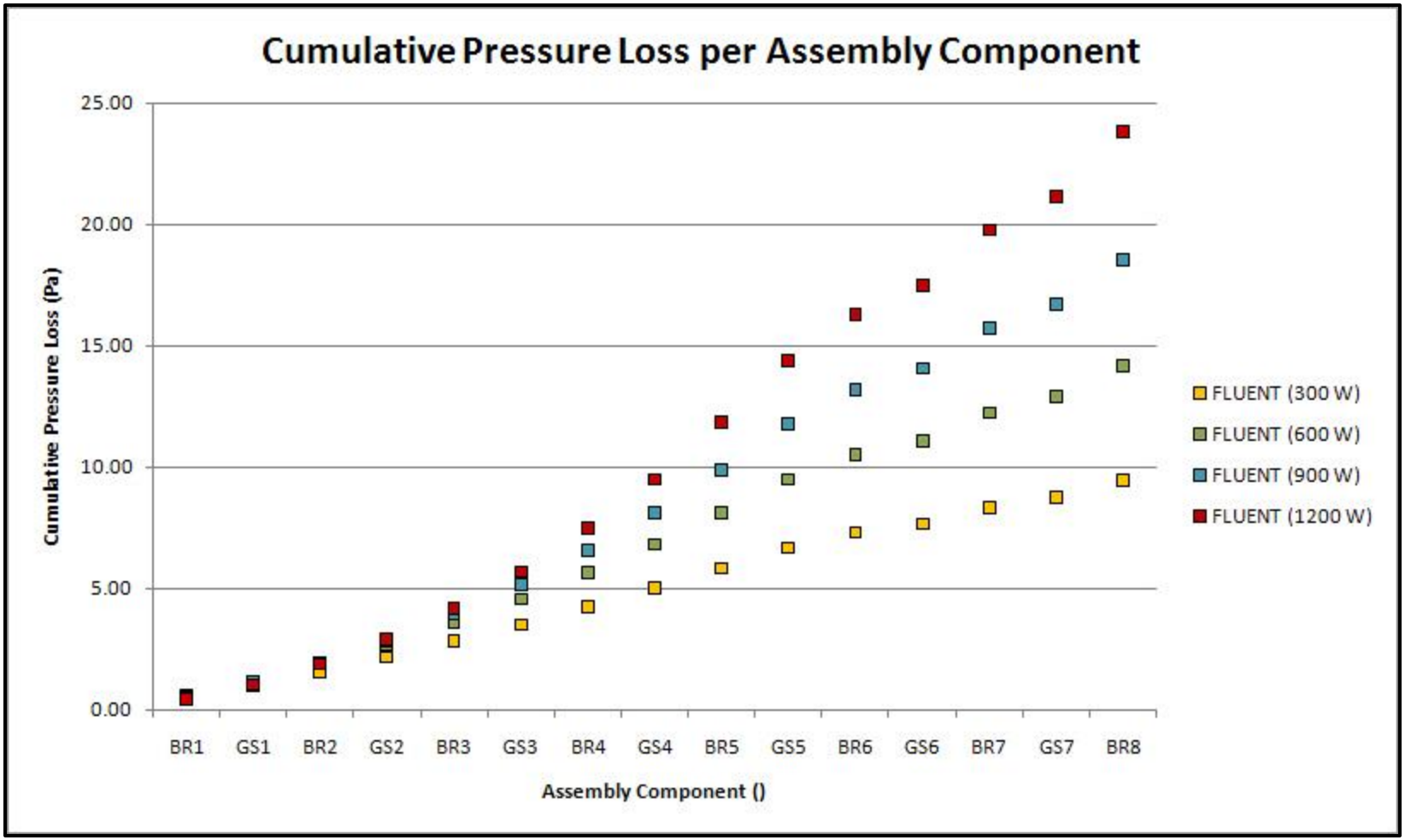


Figure 5.139:Cumulative pressure loss (MWA) per assembly component for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

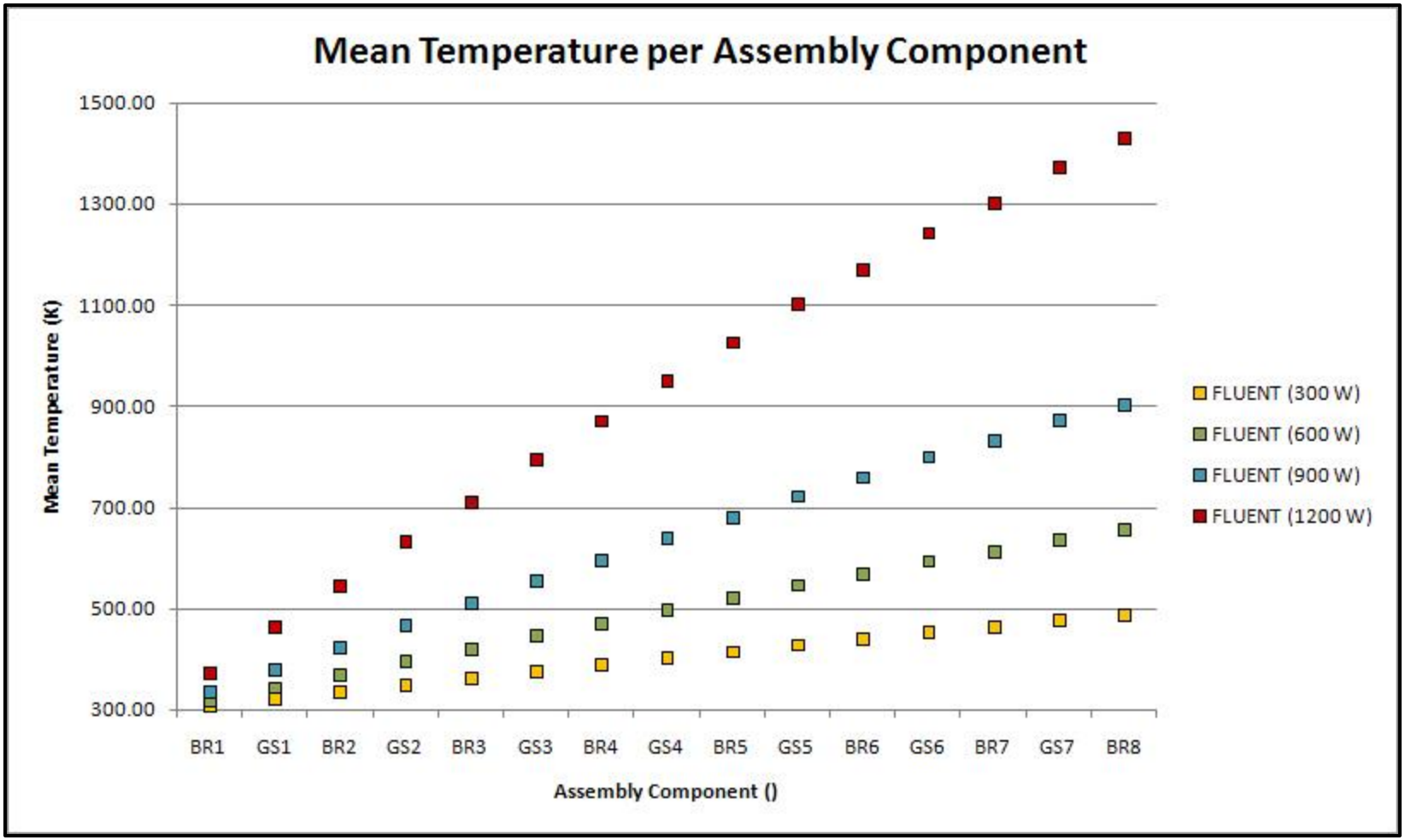


Figure 5.140:Mean temperature per assembly component for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

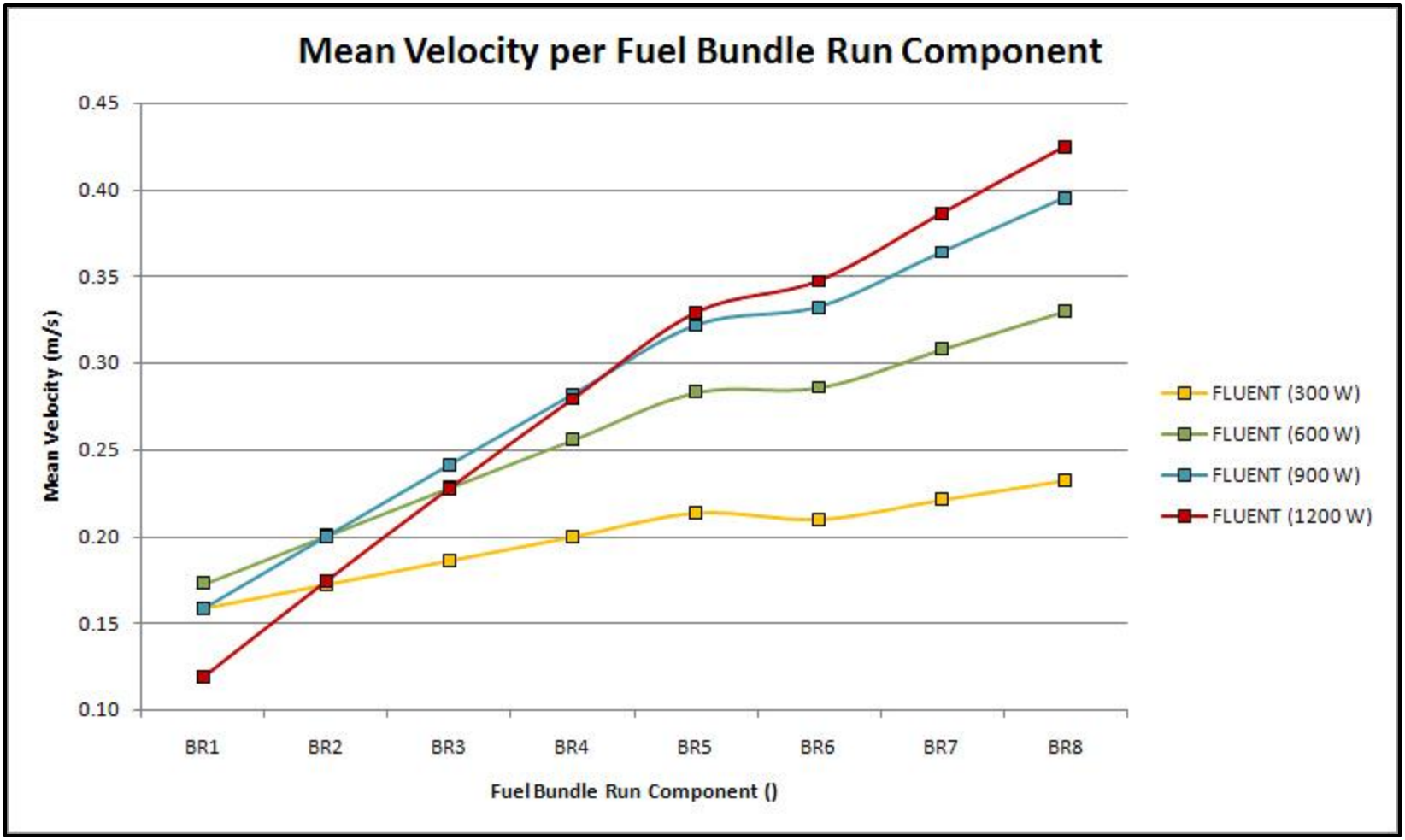


Figure 5.141:Mean velocity per fuel bundle run component for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

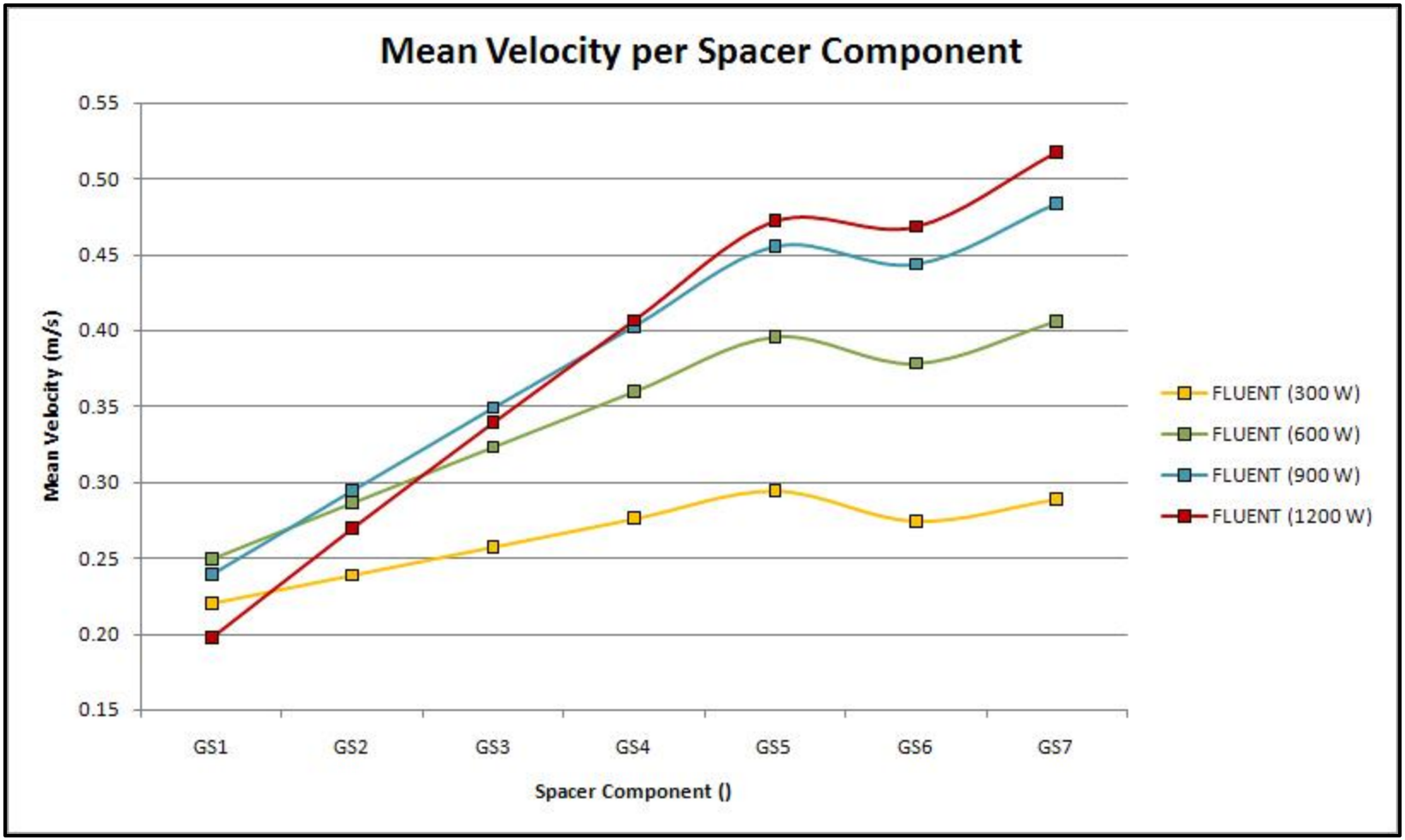


Figure 5.142:Mean velocity per spacer component for heated, buoyancy-driven flow trials using the equivalent fuel assembly model.

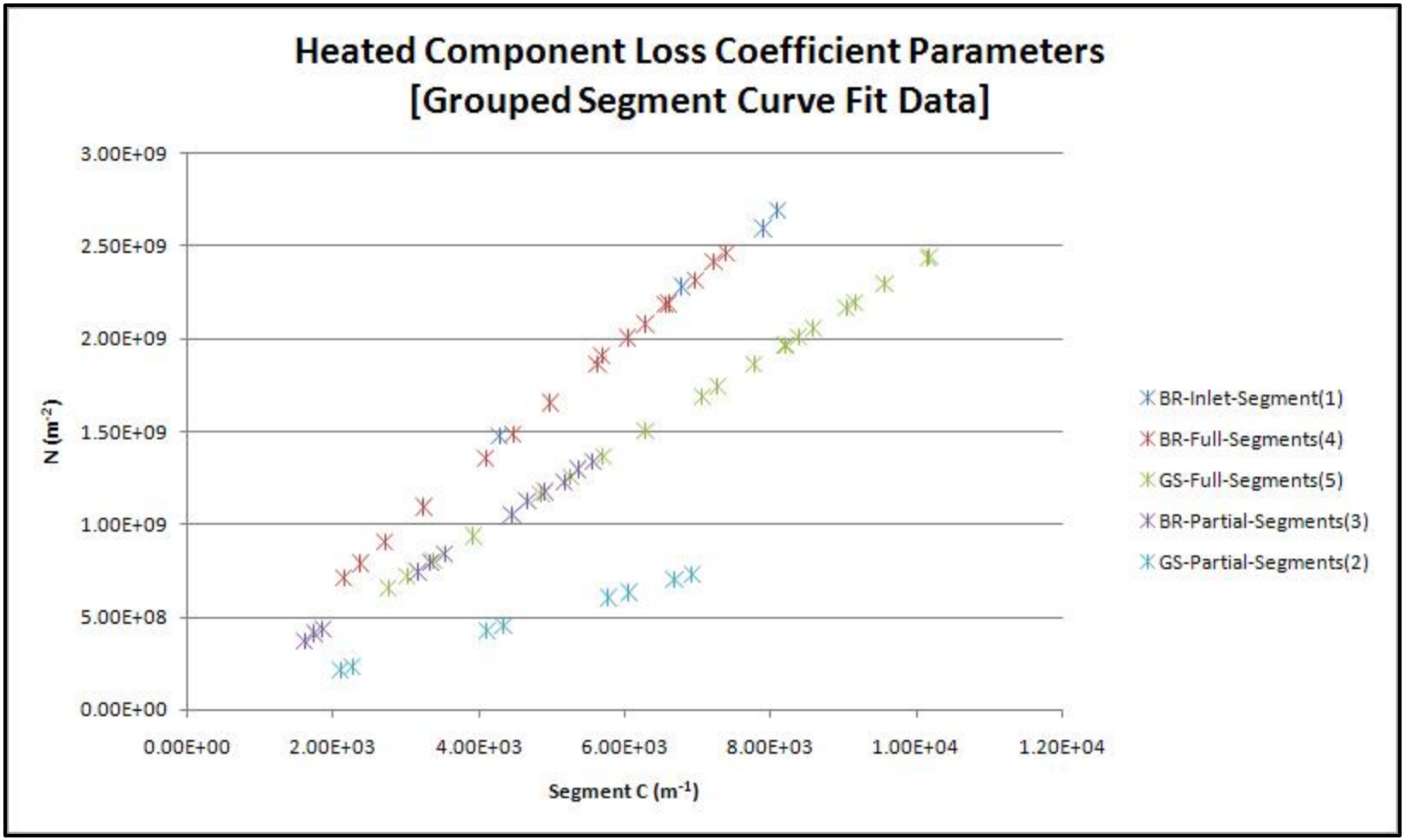


Figure 5.143:Heated component loss coefficient parameters representing the requisite curve-fit data for grouped segments used to determine the respective hydraulic loss coefficient values.

Table 5.25: Pertinent FLUENT CFD simulation results from heated, buoyancy-driven flow trials using the equivalent fuel assembly model (1 of 5).

HEATED EQUIVALENT FUEL ASSEMBLY MODEL - PERTINENT SIMULATION TRIAL RESULTS					
[Mass Flow Rate (kg/s)]		**300 W**	**600 W**	**900 W**	**1200 W**
Fuel Assembly Domain		0.00144046	0.00151808	0.00132222	0.00089970
[Exit Temperature (K)]		**300 W**	**600 W**	**900 W**	**1200 W**
Fuel Assembly Flow Outlet		499.6686	677.9746	937.8379	1488.4552
[MWA Total Pressure Loss (Pa)]		**300 W**	**600 W**	**900 W**	**1200 W**
Inlet Fuel Bundle Run 1:	BR1	0.4886213	0.5548293	0.5357549	0.4480483
Grid Spacer 1:	GS1	0.5146022	0.6152115	0.6371260	0.6053340
Fuel Bundle Run 2:	BR2	0.5746642	0.7264918	0.8018219	0.8366108
Grid Spacer 2:	GS2	0.5936779	0.7835708	0.9105529	1.0217740
Fuel Bundle Run 3:	BR3	0.6574738	0.9052898	1.0965434	1.2919392
Grid Spacer 3:	GS3	0.6766832	0.9652633	1.2133231	1.4920838
Fuel Bundle Run 4:	BR4	0.7443043	1.0974162	1.4194260	1.7944042
Grid Spacer 4:	GS4	0.7633669	1.1588014	1.5403707	2.0001116
Fuel Bundle Run 5:	BR5	0.8348732	1.3012618	1.7653681	2.3305621
Grid Spacer 5:	GS5	0.8536211	1.3631146	1.8879773	2.5376632
Fuel Bundle Run 6:	BR6	0.6212763	1.0105600	1.4159282	1.9177324
Grid Spacer 6:	GS6	0.3618724	0.5990156	0.8509664	1.1664776
Fuel Bundle Run 7:	BR7	0.6848593	1.1518204	1.6511193	2.2886068
Grid Spacer 7:	GS7	0.3946809	0.6745261	0.9796116	1.3895397
Fuel Bundle Run 8:	BR8	0.7330121	1.2695280	1.8565632	2.6719475
Entire Fuel Assembly:	BR1 → BR8	9.4975892	14.176700	18.562453	23.792835

Table 5.26: Pertinent FLUENT CFD simulation results from heated, buoyancy-driven flow trials using the equivalent fuel assembly model (2 of 5).

[Mean Velocity (m/s)]		300 W	600 W	900 W	1200 W
Inlet Fuel Bundle Run 1:	BR1	0.158080	0.172678	0.158201	0.119119
Grid Spacer 1:	GS1	0.220072	0.249046	0.239581	0.198006
Fuel Bundle Run 2:	BR2	0.172041	0.200578	0.200004	0.174617
Grid Spacer 2:	GS2	0.238665	0.286118	0.294828	0.270066
Fuel Bundle Run 3:	BR3	0.185978	0.228334	0.241271	0.228052
Grid Spacer 3:	GS3	0.257220	0.322970	0.349294	0.339563
Fuel Bundle Run 4:	BR4	0.199880	0.255894	0.281858	0.279467
Grid Spacer 4:	GS4	0.275724	0.359534	0.402827	0.406698
Fuel Bundle Run 5:	BR5	0.213740	0.283215	0.321703	0.329213
Grid Spacer 5:	GS5	0.294165	0.395759	0.455380	0.471950
Fuel Bundle Run 6:	BR6	0.210059	0.286058	0.332331	0.347514
Grid Spacer 6:	GS6	0.274234	0.378154	0.443727	0.468312
Fuel Bundle Run 7:	BR7	0.221428	0.308195	0.364126	0.386697
Grid Spacer 7:	GS7	0.288609	0.406060	0.483678	0.517408
Fuel Bundle Run 8:	BR8	0.232741	0.330115	0.395445	0.425067

Table 5.27: Pertinent FLUENT CFD simulation results from heated, buoyancy-driven flow trials using the equivalent fuel assembly model (3 of 5).

[Mean Temperature (K)]		300 W	600 W	900 W	1200 W
Inlet Fuel Bundle Run 1:	BR1	307.1983	318.4082	334.9245	370.6160
Grid Spacer 1:	GS1	320.8070	344.4800	380.4763	462.1247
Fuel Bundle Run 2:	BR2	334.3297	369.8538	423.4249	543.2896
Grid Spacer 2:	GS2	347.9110	395.7577	468.2127	630.3054
Fuel Bundle Run 3:	BR3	361.4125	421.0350	510.7903	709.5402
Grid Spacer 3:	GS3	374.9593	446.7315	554.7108	792.5037
Fuel Bundle Run 4:	BR4	388.4297	471.8541	596.7175	869.5100
Grid Spacer 4:	GS4	401.9335	497.3065	639.7250	949.1902
Fuel Bundle Run 5:	BR5	415.3636	522.2311	681.0726	1024.2838
Grid Spacer 5:	GS5	428.8163	547.4129	723.1846	1101.4811
Fuel Bundle Run 6:	BR6	440.8883	569.6979	759.8941	1167.7781
Grid Spacer 6:	GS6	453.8829	593.8757	800.0801	1240.9632
Fuel Bundle Run 7:	BR7	464.7486	613.7850	832.5950	1299.4472
Grid Spacer 7:	GS7	477.6737	637.7001	872.1152	1371.0587
Fuel Bundle Run 8:	BR8	488.4943	657.4402	904.2074	1428.3858

Table 5.28: Pertinent FLUENT CFD simulation results from heated, buoyancy-driven flow trials using the equivalent fuel assembly model (4 of 5).

[Curve Fit Parameter C (m^{-1})]		300 W	600 W	900 W	1200 W
Inlet Fuel Bundle Run 1:	BR1	7895.2369	8091.0010	6777.1964	4271.0876
Grid Spacer 1:	GS1	10175.0806	10152.3042	8205.5663	4848.3285
Fuel Bundle Run 2:	BR2	7393.2859	7217.7838	5690.8667	3248.0676
Grid Spacer 2:	GS2	9560.8170	9150.7219	7059.3660	3914.8884
Fuel Bundle Run 3:	BR3	6968.3643	6560.7084	4981.5395	2715.2783
Grid Spacer 3:	GS3	9037.0048	8380.6765	6273.2462	3369.7583
Fuel Bundle Run 4:	BR4	6603.6345	6046.6094	4477.7291	2383.8074
Grid Spacer 4:	GS4	8584.6099	7768.2758	5696.6528	3010.3573
Fuel Bundle Run 5:	BR5	6286.8524	5632.1428	4099.2740	2156.6936
Grid Spacer 5:	GS5	8189.5916	7268.2191	5253.8639	2753.7983
Fuel Bundle Run 6:	BR6	5575.3847	4912.0698	3533.6305	1847.9645
Grid Spacer 6:	GS6	6924.9042	6055.7928	4335.3414	2260.7094
Fuel Bundle Run 7:	BR7	5369.5849	4670.3299	3333.2948	1733.7832
Grid Spacer 7:	GS7	6678.4621	5772.8117	4105.6897	2095.7336
Fuel Bundle Run 8:	BR8	5183.9994	4460.7288	3165.6944	1599.3798

Table 5.29: Pertinent FLUENT CFD simulation results from heated, buoyancy-driven flow trials using the equivalent fuel assembly model (5 of 5).

[Curve Fit Parameter N (m^{-2})]		300 W	600 W	900 W	1200 W
Inlet Fuel Bundle Run 1:	BR1	2.5918153E+09	2.6847790E+09	2.2794477E+09	1.4777465E+09
Grid Spacer 1:	GS1	2.4428560E+09	2.4377706E+09	1.9683369E+09	1.1609581E+09
Fuel Bundle Run 2:	BR2	2.4560331E+09	2.4084479E+09	1.9026924E+09	1.0885902E+09
Grid Spacer 2:	GS2	2.2943895E+09	2.1956483E+09	1.6919127E+09	9.3678345E+08
Fuel Bundle Run 3:	BR3	2.3091759E+09	2.1782101E+09	1.6528010E+09	8.9953085E+08
Grid Spacer 3:	GS3	2.1679269E+09	2.0098356E+09	1.5027237E+09	8.0609467E+08
Fuel Bundle Run 4:	BR4	2.1843601E+09	2.0013188E+09	1.4796871E+09	7.8579685E+08
Grid Spacer 4:	GS4	2.0588024E+09	1.8622471E+09	1.3641308E+09	7.2000195E+08
Fuel Bundle Run 5:	BR5	2.0767235E+09	1.8602793E+09	1.3513443E+09	7.0915235E+08
Grid Spacer 5:	GS5	1.9638701E+09	1.7421140E+09	1.2580265E+09	6.5874569E+08
Fuel Bundle Run 6:	BR6	1.3357145E+09	1.1750681E+09	8.4203146E+08	4.3835386E+08
Grid Spacer 6:	GS6	7.2496657E+08	6.3149840E+08	4.4987786E+08	2.3350063E+08
Fuel Bundle Run 7:	BR7	1.2956042E+09	1.1237770E+09	7.9742533E+08	4.1382052E+08
Grid Spacer 7:	GS7	6.9878996E+08	6.0178967E+08	4.2610967E+08	2.1635562E+08
Fuel Bundle Run 8:	BR8	1.2296717E+09	1.0549069E+09	7.4469347E+08	3.7402062E+08

Table 5.30: Hydraulic loss coefficient results from heated, buoyancy-driven flow trials for grouped (flow path) segments using the equivalent fuel assembly model.

EQUIVALENT FUEL ASSEMBLY MODEL - TABLECURVE 2D CURVE FIT DATA - HEATED HYDRAULIC LOSS COEFFICIENTS					
[BR-Inlet-Segment]	a1:	a2:	R^2:	SLAM:	k:
300 - 1200 (W)	364473.660	-4.268716	0.9993	109.1748	-4.268716
[BR-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
300 - 1200 (W)	329865.863	0.300386	0.9998	98.8083	0.300386
[GS-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
300 - 1200 (W)	238762.937	0.128565	1.0000	66.2418	0.128565
[BR-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
300 - 1200 (W)	234491.975	0.930330	0.9996	98.0355	0.930330
[GS-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
300 - 1200 (W)	102446.932	0.318634	1.0000	43.3600	0.318634

5.3.4 Final Comparison of Results

Due to deviating boundary condition specifications and nature of inflow developments intrinsic between the unheated and heated model setups at the domain inlet, notable differences are expected among the resulting values for hydraulic loss coefficients associated with the inlet fully-populated bundle run segment. The negative value obtained for the heated form loss coefficient attributed to this inlet segment is of preliminary concern that necessitates further examination. Initially, the cause is thought to stem from one or more sources including widely divergent thermal and momentum boundary layer developments as well as flow-field properties across the segment, a limited sampling of fitted data points, and inadequate representation of the actual domain flow field by using a single set of averaged property values over the entire segment volume. Otherwise, aside from form loss coefficients for bundle run segments, compared hydraulic loss coefficients of remaining fuel assembly components from unheated and heated trials are well within an acceptable level of agreement as shown by the final conglomeration of resulting loss coefficient values in Table 5.31 below, which is also referenced throughout the remainder of this closing research summary.

Table 5.31: Final comparison of hydraulic loss coefficient results for unheated, pressure-driven and heated, buoyancy-driven airflow trials using the equivalent fuel assembly model.

EQUIVALENT FUEL ASSEMBLY MODEL - COMPARISON - UNHEATED & HEATED HYDRAULIC LOSS COEFFICIENTS

[Fitted-Reference-Equations]

Unheated:

$$\Delta P_{total} = a_2 V^2 + a_1 V$$

where, $S_{LAM} = 2a_1\left(\frac{D_h^2}{L\mu}\right)$ $\sum k = \frac{2a_2}{\rho}$

Heated:

$$N = a_1(C) + a_2(C^2)$$

where, $N = \frac{2\Delta P_{total}\rho_m}{\mu_m^2}$ $C = \frac{\rho_m V_m}{\mu_m}$ $S_{LAM} = \frac{a_1 D_h^2}{L}$ $\sum k = a_2$

[BR-Inlet-Segment]	a1:	a2:	R^2:	SLAM:	k:
Unheated: 200 - 600 (slpm)	2.941034	0.030252	1.0000	96.7569	0.061650
Heated: 300 - 1200 (watt)	364473.660	-4.268716	0.9993	109.1748	-4.268716
[BR-Fully-Populated-Segments]	**a1:**	**a2:**	**R^2:**	**SLAM:**	**k:**
Unheated: 200 - 600 (slpm)	2.969000	0.267182	1.0000	97.6770	0.544484
Heated: 300 - 1200 (watt)	329865.863	0.300386	0.9998	98.8083	0.300386
[GS-Fully-Populated-Segments]	**a1:**	**a2:**	**R^2:**	**SLAM:**	**k:**
Unheated: 200 - 600 (slpm)	2.169390	0.075111	1.0000	66.1042	0.153067
Heated: 300 - 1200 (watt)	238762.937	0.128565	1.0000	66.2418	0.128565
[BR-Partially-Populated-Segments]	**a1:**	**a2:**	**R^2:**	**SLAM:**	**k:**
Unheated: 200 - 600 (slpm)	2.200103	0.253063	0.9993	101.0240	0.515710
Heated: 300 - 1200 (watt)	234491.975	0.930330	0.9996	98.0355	0.930330
[GS-Partially-Populated-Segments]	**a1:**	**a2:**	**R^2:**	**SLAM:**	**k:**
Unheated: 200 - 600 (slpm)	0.934320	0.151016	1.0000	43.4322	0.307751
Heated: 300 - 1200 (watt)	102446.932	0.318634	1.0000	43.3600	0.318634

A potential explanation for the discrepancy between unheated and heated form loss coefficients for bundle run segments arises from the observation of an apparent systematic relationship among loss coefficient differences in all compared segments including those with reasonably similar solution values. In other words, for each unheated and heated segment comparison above, the difference between viscous loss coefficients is offset to some degree by an inverse difference in the form loss coefficient values compared for that same fuel assembly segment. This systematic influence is initially attributed to the slightly differing approach used to obtain requisite heated flow-field data and calculate curve-fit results, where averaging of certain characteristic velocity and property values over the segment flow volume is necessary.

As expected, this influence imparts a greater consequence across longer length segments such as fuel bundle runs and ultimately a larger impact on the systematic difference observed between unheated and heated form loss coefficients. In conjunction, the span of possible flow-field solutions and range of corresponding data values comprising the curve-fit evaluations inherently varies between the unheated and heated set of computational trials conducted. As a result, linear or nonlinear components of unheated and heated data sets compared for a particular segment are likely to incur influences of varying extent based on the respective range of evaluated solutions in each spanning set of curve-fit data, which also follows the observation of a systematic relationship among contrasted loss coefficient differences.

Therefore, based on this initial research effort and assessment outlined above, the non-geometric parameters that potentially affect the specific values of and/or differences between unheated and heated hydraulic loss coefficients characterizing a traversing assembly flow field include the following: (1) inlet boundary condition and profile specification; (2) interaction of thermal and momentum boundary layer development on

flow field across inlet fuel bundle; and (3) resolution of evaluated segment lengths where actual flow-field velocity and property values are represented as single volume averaged quantities for the entire segment. Furthermore, it is pertinent to note that form loss coefficients from both unheated and heated computational trials are appreciably reduced in comparison to those obtained from SNL experiments of unheated, forced assembly airflows; the only exception pertains to partially-populated bundle run segments under heated trial conditions. Assuming these larger SNL values accurately represent the form loss coefficients quantifying minor losses in unheated flows across the actual fuel assembly geometry, considerably increased deviations are certainly plausible between unheated and heated form loss coefficients characterizing the real fuel assembly structure than discerned above for an equivalent domain geometry and model.

If such a realization is determined as a consequential effect of more-pronounced inertial losses associated with an actual assembly flow volume, a possibility remains that increased discrepancies are governed by the development of variant flow-field features as well as resulting solutions that do not coincide with similar characteristic attributes exhibited by the equivalent model results. Subsequently, an explanation for these increased discrepancies in terms of a systematic relationship among coefficient differences and substantiating arguments akin to those describing results from the equivalent assembly model is likely discerned as inadequate for this prescribed situation. Obviously, a definitive assessment of the correlations between unheated and heated loss coefficients necessitates further research efforts with regard to larger-scale computational modeling and/or physical experiments of heated, buoyancy-driven airflows across an actual fuel assembly structure.

In summary, this research endeavor successfully demonstrates the application of FLUENT CFD software to provide beneficial results through an alternative means of

indirect analysis involving simplified domain models and ultimately promoting a greater understanding of the expected flow-field responses within SFP assemblies under complete LOCA conditions. Although several discrepancies are clearly noted, expected deviations as well as concerns associated with unheated and heated loss coefficients for the inlet fully-populated bundle run segment are addressed by logical considerations of limited congruency among model setups, flow-field development, and data range, which preclude the indication that divergent unheated and heated loss coefficients absolutely subsist. Similarly, variations between unheated and heated form loss coefficients for bundle run segments are rationally explained in terms of an apparent systematic relationship among loss coefficient differences for all compared segments including several substantiating arguments of distinct interest.

Therefore, based on the general agreement between hydraulic loss coefficients from unheated and heated airflow trials employing the equivalent assembly model in conjunction with elucidatory explanations of aforementioned discrepancies, a practical similarity is sufficiently established to conclude ascertainment of reasonable approximations for heated, buoyancy-driven airflow responses across a fuel assembly by implementing loss coefficients describing unheated, pressure-driven assembly airflows. Further evidence supporting this conclusion is revealed throughout a number of validation assessments and sensitivity studies, which incorporate these loss coefficient values derived from the equivalent assembly model in developing a MATLAB buoyancy-driven flow program. These considerations are the focus of research efforts and discussions outlined in the ensuing chapter.

Chapter 6: MATLAB Program for Buoyancy-Driven Flow

6.1 INTRODUCTION

The primary objective of the following research is directed toward the development of a simple, computationally efficient MATLAB program for predicting the buoyancy-driven air mass flow rate and temperature distribution across a prototypical 9x9 BWR fuel assembly in a SFP under complete LOCA conditions. Although high-fidelity CFD models are capable of providing accurate solutions to analyses of very complex systems with a minimal number of required assumptions, the method incurs a significant cost in the form of model development effort, dedication of extensive computing resources, and substantial CPU-hour time increments for completion. Therefore, a frequently sought benefit of conducting a limited number of intense CFD computations is the attainment of critical parameter values and/or data for empirical/constitutive correlations necessary in the development of (or as input to) more computationally efficient, inexpensive model alternatives that afford relatively accurate solutions to the same system under varying perturbations.

Evaluating the potential application of such a benefit for the development and implementation of a more efficient, simplified modeling approach that facilitates the resolution of thermal hydraulic responses within a BWR fuel assembly under LOCA conditions is central to the ensuing research endeavor. Moreover, examinations focus on utilizing the available flow-field data in a manner similarly dictated by SNL for exploiting their hydraulic analysis results from the acquired experimental measurements. Unfortunately, as formerly mentioned, drawing upon a large production-type mesh model for resolving flow fields through an entire BWR fuel assembly necessitates further model

development and accuracy assessment than currently completed/achievable at this stage of research (see Section 5.2).

As a result, using only a simplified equivalent representation of assembly domain geometry, a two-part alternative method is outlined in order to examine the capability of FLUENT CFD models to impart critical solution parameters requisite for implementing/developing an effective secondary model and determine whether or not a reasonable advantage is afforded by the derived application for related system analyses. This two-part method is initially introduced in Section 5.3 above, which also includes concluding discussions and results stemming from successful completion of the first constituent part to this method. Accordingly, the second aspect of the two-part method is subsequently addressed by research efforts, considerations, and findings disclosed throughout this chapter. Thus, aside from a more definitive evaluation of the congruency between flow-field characteristics dependent upon the entire domain volume in unheated, forced flows and heated, buoyancy-driven flows through a hydraulic equivalent mockup model of a BWR fuel assembly, the CFD solutions from aforementioned simulations in Section 5.3 are additionally incorporated into the ensuing research as follows:

1. Provide supporting CFD solution data for necessary parameters and constitutive/phenomenological/empirical equation inputs as required in the initial development of the MATLAB buoyancy-driven flow program.
2. Evaluate potential (future) capability of high-fidelity, large production-type mesh models encompassing the entire fuel assembly to furnish CFD simulation results that include unknown critical parameter values/data governing the thermal hydraulic flow-field response under complete LOCA conditions.

3. As a complementary component to the preceding point, afford a means of validating the MATLAB buoyancy-driven flow program results; likewise, exhibiting reasonable agreement between these MATLAB program results and CFD simulation solutions from Section 5.3 also imparts evidence supporting the capability of FLUENT CFD flow models to accurately supply the essential hydraulic parameter values for use in such applications.

In order to complete the above assessments successfully, the developed MATLAB program code must function as a suitable, simplified model in which similar analyses are conducted and comparable results are easily obtained using the appropriate form of input values acquired from the CFD solutions. Unfortunately, consideration of existing program models/code structures intended for somewhat similar application (*i.e.* SFPT and GFLOW) reveal a number of inherent deficiencies that render the codes unsuitable for use in a comparative analysis. Furthermore, the potential to modify the existing codes and/or development platforms for accurately predicting buoyancy-induced mass flow rates associated with varying levels of fuel decay heat input rate in a BWR fuel assembly is essentially absent. Nonetheless, the identification of a well-validated analytical method used in the modeling of systems possessing similarly observed physical phenomena that serves as a base skeletal outline for constructing the specific programming code, is a precursory task of primary importance. Initial deliberation of preferred program attributes concluded with an additional constraint for implementing a solution methodology that is also conducive to future subroutine appendages incorporating physical thermal radiation and/or combined convection-conduction heat transfer mechanisms if desired.

An extensive review of related topics in the literature reveals a strikingly analogous correlation between the physical phenomena associated with complete LOCA conditions in a SFP and the normal operation environment in solar energy building technologies. In particular, recent research efforts considering the effects of most hydraulic losses and developing inlet flow profiles on design analyses of PV cooling ducts yielded a single loop solution method (Brinkworth *et al.*, 2000) and several subsequent model variations (Brinkworth, 2000; Brinkworth and Sandberg, 2005), explicitly derived for application in the estimation of buoyancy-induced flow rates in ventilated ducts with a net rate of internal heat gain. The simple, computationally efficient 'single loop' analysis with unique 'stratification parameter' for describing axial heat flux distribution (Brinkworth *et al.*, 2000), provides a sufficient one-dimensional analytical base method for the development of a specific buoyant flow model and program code applicable to the prescribed BWR fuel assembly in complete LOCA scenarios.

However, significant tailoring of thermal hydraulic mechanisms evaluated and the actual solution procedure implemented (in comparison to the presented research in above-mentioned references) is required due to a diverse interest in explicit physics modeled between the systems considered. The final derivation of the thermal hydraulic governing equations and solution procedure incorporated into the MATLAB buoyancy-driven flow program are topics of primary discussion in forthcoming sections of this chapter. Nevertheless, prior to the explanation of relevant theory, a number of intricacies detailing the general program development warrant a brief introduction including several aspects of applied assumptions, intended functionality, and limiting attributes associated with the current base version of program code. These are addressed in a bulleted point summary as follows:

- Unless otherwise noted, the general assumptions, boundary conditions/types, and respective rationale for each are consistent with those expressed in Section 5.3 above (especially with regard to the heated simulation trials in Section 5.3.3) concerning the simplified CFD mesh model representing an equivalent hydraulic scale fuel assembly domain.
- The single loop method of analysis evaluates a control volume of the entire flow domain comprised of connected flow path segments, which also constitute separate computational control subvolumes.
- At present, the lower and upper tie plate components are not specifically included in the flow resistance calculations, but may potentially be accounted for with reasonable reflection through indirect modification of input values for the inlet and outlet form loss coefficients. Ease of direct incorporation into future code revisions is maintained by implementing a modular solution formulation.
- A similar form of method utilized in the heated flow simulations of Section 5.3.3, termed the incompressible ideal gas law approach in FLUENT CFD software, is applied for the density relationship and related calculations of the flow regime. Validity is based on pressure variations within the flow domain that are small enough for a fully incompressible flow (*i.e.* density is dependent on the operating pressure and not on the local relative pressure field); however, the relationship between density and temperature from the ideal gas law is necessary in accounting for the body forces driving the fluid airflow.
- Converged solutions are for steady-state conditions in which the mass flow rate is conserved and constant along the entire single loop flow path. Key results from

the program output include the overall air mass flow rate as well as the temperature, velocity, and pressure loss for each flow path segment.

- Flow losses and specific heat transfer mechanisms in return flows traveling along downcomer or SFP rack/base-plate/boundary walls are not considered in the current base program, but the code is structured along with the solution method such that only a few modifications are necessary to incorporate these additional loss/heat transfer contributions into the return flows. In other words, the portion of the single loop flow path that returns the exiting assembly flow back to the inlet is an infinite sink offering no resistance to flow, but serves to re-establish conditions within loop to initial/inlet pressure and temperature values.
- Similar forms for the major and minor pressure loss equations as well as the constituent hydraulic loss coefficients, which are defined and determined according to Chapters 3 and 5, are implemented into the program code to maintain conformity between specified coefficient input values and ultimately simplify the comparison of results.
- The net (decay) heat input rate of assembly fuel is balanced at steady-state conditions with a heat extraction rate from the system domain attributable to buoyancy-induced airflow through the fuel assembly apparatus. This is a reasonable assumption for a steady-state analysis of a worst-case scenario whereby adjacent fuel assemblies in the same and/or surrounding SFP rack(s) possess near-equivalent levels of decay power.
- Such conditions minimize heat removal rate from the system by means other than internal heat gain absorbed by air since neighboring outer surfaces have similar temperatures at a given axial height, which precludes any significant radiative or conductive heat transfer.

- Specifically, the close proximity between fuel assembly/SFP rack outer walls restrains the view factors for radiative heat transfer from wall segments at higher temperature (higher axial height) to those of adjacent fuel assembly/SFP rack outer walls at lower temperature (lower axial height). Heat transfer from the fuel assembly to lower supporting SFP rack structures (*i.e.* base-plate, footing, or ground floor) by conduction is also restricted by the limited path (cross-sectional area) of solid material contact.
- The one-dimensional aspect of the heat removal rate by airflow in the single loop method of analysis is also a reasonable approximation to the steady-state solution for similar reasons in terms of axial radiative and conductive heat transfer (these rates are comparatively small, even with lower air mass flow rates, given the significantly larger surface area for convection).
- Furthermore, recent research concerning intra-assembly radiative heat transfer in the radial direction (*i.e.* normal to fluid flow direction) for similar BWR fuel assemblies under complete LOCA conditions, demonstrated that the effect of thermal radiation heat transfer at axial height locations is an increased pacification of the temperature gradient over the assembly cross-section (Reinert, 2004; Artnak *et al.*, 2005; Artnak, 2008).
- As a result, the indirect effect of intra-assembly radiative heat transfer is a maximization of convective heat transfer to the airflow by redistributing the fuel rod surface temperatures toward uniformity over the domain cross-section. Clearly, this effect supports the one-dimensional heat transfer approximation for a steady-state analysis of a worst-case scenario.
- Despite reasonable justifications for underlying assumptions of net heat removal rate by airflow only and sufficient approximation by one-dimensional analysis of

the heat transfer rate, the code is again structured along with the solution method such that future revisions for incorporating direct calculation of one or more specific modes of heat transfer (*i.e.* radiation, conduction, and convection) within/between one or more different solid/fluid regions are plausible with relatively straightforward modifications to existing code. This is conceivable due to the simplicity of the heat transfer rate expressed in the energy balance equation of the current program code.

6.2 RELEVANT THEORY

The relevant theory incorporated throughout formulation of the governing equations used in the MATLAB buoyancy-driven flow program is briefly introduced in the following section. This information is essential for a general understanding of the actual code development and numerical solution procedure outlined in the ensuing discussions of this chapter. Initially, the fundamental aspects of the single loop analysis and physics associated with the method of predicting the thermal hydraulic response of the control volume domain is presented in the most straightforward manner by considering an extremely basic geometrical representation of an assembly control volume encompassing a single flow path segment. A schematic of the setup with accompanying description of principal characteristics and related parameters is illustrated in Figure 6.1 below.

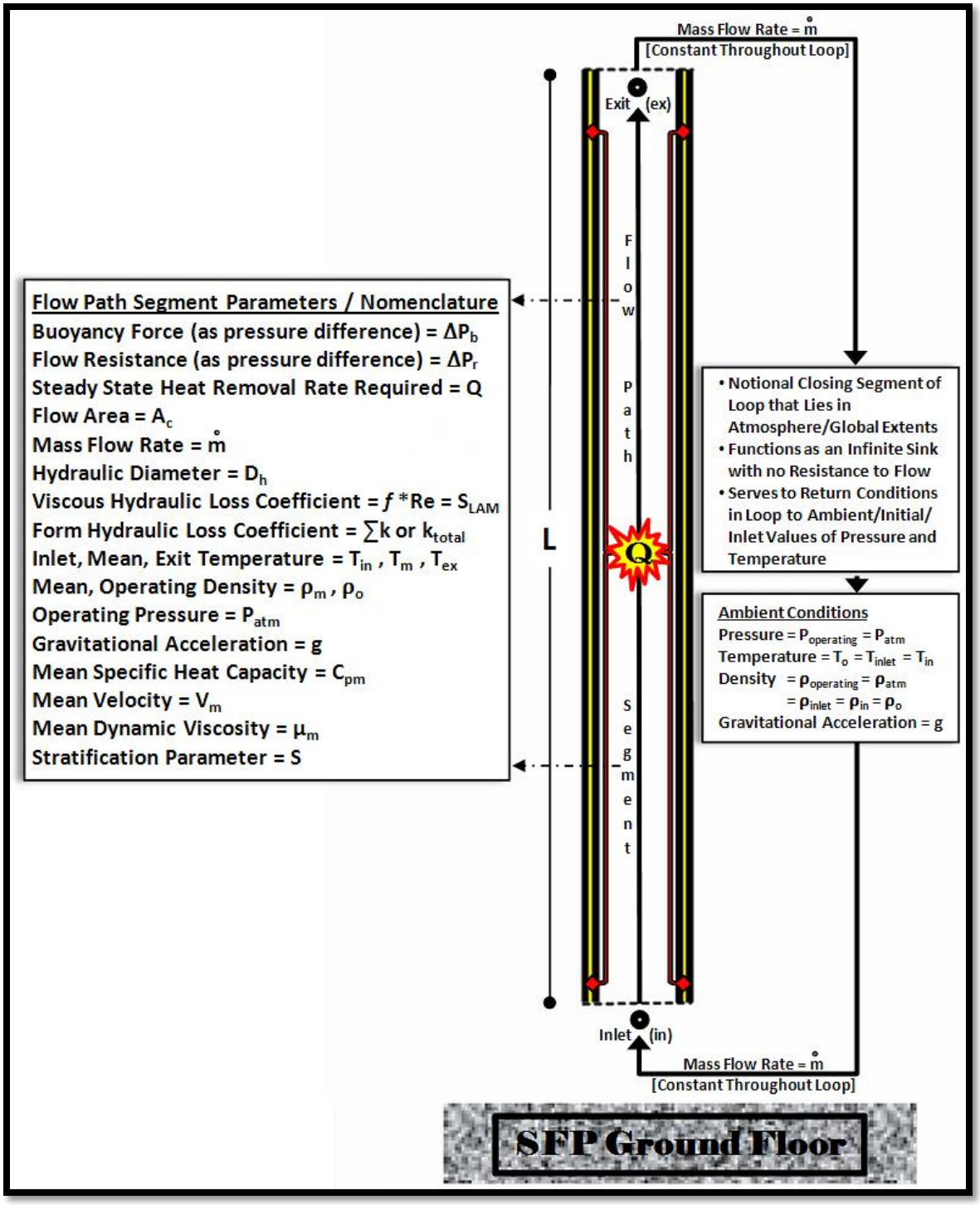


Figure 6.1: Schematic of simplified example representing an assembly control volume with parameter descriptions for initial consideration of single loop analysis.

Applying the general considerations outlined at the end of Section 6.1 to the simplified example schematic and parameters detailed in Figure 6.1 above, the single loop analysis of the assembly control volume asserts that resistances to flow around the loop are balanced by equal forces driving the airflow due to buoyancy. In terms of pressure differences, this provides a constraint for steady-state conditions around the control volume loop as follows:

$$\Delta P_r - \Delta P_b = 0 \tag{6.1}$$

where, the total resistance to flow (ΔP_r) is the sum of major and minor pressure losses determined using similar forms of the constitutive/phenomenological equations from Chapters 3 and 5 above, and the driving buoyancy pressure differential (ΔP_b) is ascertained from the gravitational force difference between the opposing assembly internal and ambient external air columns along the domain length.

Due to the breadth of prior consideration with regard to the widely varying equation forms for the major and minor pressure losses applied in the aforementioned research, a representative equation for the total resistance to flow is initially detailed. Exercising parameter definitions and notations illustrated in Figure 6.1, the sum of major and minor pressure losses constituting the total resistance across the flow path segment is given by

$$\Delta P_r = f\left(\frac{L}{D_h}\right)\left(\frac{\rho_m V_m^2}{2}\right) + \sum k\left(\frac{\rho_m V_m^2}{2}\right) \tag{6.2}$$

where, the friction factor (f) in terms of the Reynolds number (Re_{Dh}) is expressed as

$$f = \frac{S_{LAM}}{Re_{Dh}} = \frac{S_{LAM}\mu_m}{\rho_m V_m D_h} \tag{6.3}$$

Substituting Equation (6.3) into Equation (6.2) for the friction factor (*f*) and regrouping, Equation (6.2) becomes

$$\Delta P_r = \frac{S_{LAM}\mu_m L}{2D_h^2}(V_m) + \frac{\sum k}{2}(\rho_m V_m^2) \tag{6.4}$$

Since the constant, steady-state value for the air mass flow rate ($\dot{m}$) is of primary interest, a slightly modified expression for the mass flow rate in terms of the parameters stated in Equation (6.4) is necessary. However, this is conceived with minimal effort as follows

$$\dot{m} = \rho_m V_m A_c \Longrightarrow \rho_m V_m = \frac{\dot{m}}{A_c} \tag{6.5}$$

Hence, using the indirect relationship from Equation (6.5) and rearranging, the desired form of Equation (6.4) for the total resistance to flow in terms of the mass flow rate parameter is finally obtained as

$$\Delta P_r = \left(\frac{\sum k}{2\rho_m A_c^2}\right)\dot{m}^2 + \left(\frac{S_{LAM}\mu_m L}{2D_h^2 \rho_m A_c}\right)\dot{m} \tag{6.6}$$

Equation (6.6) represents the base general equation for the total resistance to flow around the control volume loop and serves as one of the fundamental starting point equations in subsequent derivations.

The elementary governing equation form characterizing the summed contribution of buoyancy forces around the control volume loop, expressed in terms of relevant pressure differences, is derived in a rather straightforward fashion at the outset of consideration. Again, using the nomenclature shown in Figure 6.1, the driving buoyancy pressure differential (ΔP_b) resulting from the gravitational force difference between the opposing assembly internal and ambient external air columns along the domain length is given by

$$\Delta P_b = gL(\rho_o - \rho_m) \tag{6.7}$$

For reasons that become apparent in latter development of the program code, a relationship between the airflow density and temperature is applied using the ideal gas law in order to express Equation (6.7) in a slightly modified form. Hence, using the ideal gas law equation for the density of air as

$$\rho = \frac{P_{atm}}{R_{gas,atm}T} \tag{6.8}$$

a statement of the differential density ($\rho_o - \rho_m$) becomes

$$(\rho_o - \rho_m) = \frac{P_{atm}}{R_{gas,atm}T_{in}} - \frac{P_{atm}}{R_{gas,atm}T_m} = \rho_o\left(1 - \frac{T_{in}}{T_m}\right) = \rho_o\left[\frac{(T_m - T_{in})}{T_m}\right] \tag{6.9}$$

Substituting Equation (6.9) into Equation (6.7), yields the buoyancy pressure differential in the following new equation form

$$\Delta P_b = gL\rho_o \left[\frac{(T_m - T_{in})}{T_m}\right] \tag{6.10}$$

As with Equation (6.6), it is prudent to write Equation (6.10) in terms of the mass flow rate parameter ($\dot{m}$) since this is the solution value of greatest interest. However, a second formulation with regard to the conservation of energy is essential in achieving this desired form and is considered accordingly in the subsequent discussion.

The final governing expression of physics represented in the above system is derived from the simple necessity to ensure the conservation of energy across the entire control volume loop. Specifically, based on the previously stated assumptions, this is a system stipulation that requires a balance between the net decay heat input rate of the fuel and the heat removal rate from the system due to the buoyancy-induced airflow rate at steady state. Therefore, the energy balance in terms of the heat removal rate required (Q) is simply a statement of the internal rate of heat gain absorbed by air at steady-state flow, which is modestly expressed using the nomenclature defined in Figure 6.1 as

$$Q = \dot{m}C_{Pm}(T_{ex} - T_{in}) \tag{6.11}$$

Addition of the unknown exit air temperature parameter (T_{ex}) and the lack of a prescribed correlation for the mean air temperature (needed to obtain mean airflow properties), is an undesired consequence presented by the above form of Equation (6.11) that requires further deliberation.

A critical step in the formulation of a solution procedure for the described problem is establishing a reasonably justifiable approximation for linking the mean air temperature with the temperature distribution across a flow path segment. Fortunately,

recent research involving a number of buoyancy-driven, high-aspect-ratio duct flows in solar energy building technologies produced a simple solution to this similarly encountered problem in the form of a defined 'stratification parameter' term; use of this form is well validated by computational and experimental modeling studies of equivalent flow phenomena in heated ducts (Yang *et al.*, 1996; Brinkworth *et al.*, 2000; Brinkworth, 2000; Brinkworth and Sandberg, 2005). The stratification parameter (*S*) is dependent upon the heat flux distribution along the duct length and quantified by the temperature profile that develops as a consequence of this prescribed heat flux.

Ultimately, the temperature distribution traversing the domain defines the mean temperature within the channel, which Brinkworth *et al.* (2000) approximate through application of the stratification parameter (*S*) as follows:

$$T_m = T_{in} + S(T_{ex} - T_{in}); \quad for\ 0 \leq S \leq 1 \tag{6.12}$$

The capability of this equation form to represent a number of different situations and associated temperature distributions is exemplified by consideration of several examples: (1) a naturally ventilated room modeled as fully mixed in which the mean temperature is the exit temperature, (*S*) takes on the value of 1; (2) an ideal/perfect displacement ventilation operation in which the mean temperature is the inlet temperature, (*S*) $\rightarrow$ 0 is expected; (3) a solar air collector in which the temperature distribution is exponential, 0.5 $<$ (*S*) $<$ 1 is appropriate; (4) a non-ideal displacement ventilation operation, 0 $<$ (*S*) $<$ 0.5 is expected; and (5) a PV cooling duct in which the temperature distribution is nearly linear, (*S*) takes on the value of 0.5 (Brinkworth *et al.*, 2000). In Figure 6.2 below, the attributed shape of the temperature distributions as a function of the stratification parameter (S) is illustrated for several of these examples.

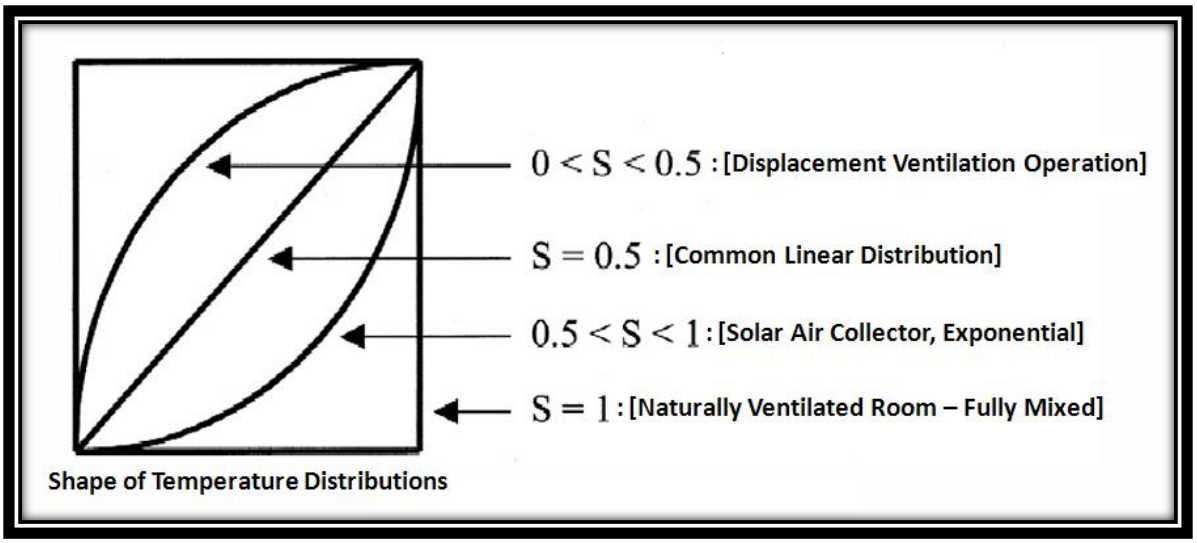


Figure 6.2: Shape of temperature distributions as a function of the stratification parameter (*S*) value; adapted in part from (Brinkworth *et al.*, 2000).

Therefore, with knowledge of the prescribed heat flux distribution along the domain length, a reasonable estimation of the stratification parameter value (the sensitivity of which is evaluated in Section 6.4.2 below) is permissible based on the expected shape of the resulting temperature distribution. Using the defining relationship for the stratification parameter (*S*) given by Equation (6.12) to express the temperature differential across the entire flow path segment ($T_{ex} - T_{in}$), Equation (6.11) is subsequently rewritten as

$$Q = \dot{m} C_{Pm} \left[\frac{(T_m - T_{in})}{S} \right] \tag{6.13}$$

An expression for the mean temperature rise in the flow path segment ($T_m - T_{in}$) is obtained with slight rearrangement of Equation (6.13), which yields

$$(T_m - T_{in}) = \frac{QS}{\dot{m}C_{Pm}} \tag{6.14}$$

Thus, by substituting Equation (6.14) into Equation (6.10) for the mean temperature rise $(T_m - T_{in})$, a form of Equation (6.10) in terms of the desired mass flow rate is given by

$$\Delta P_b = gL\rho_o \left(\frac{QS}{\dot{m}C_{Pm}T_m}\right) \tag{6.15}$$

Equation (6.15) represents the basic general equation for the summed driving buoyancy pressure differential (ΔP_b) around the control volume loop, written in the desired format for any subsequent, future derivations of actual equations implemented into the development of program code.

Achieving expressions for both the total resistance to flow (ΔP_r) and buoyancy pressure differential (ΔP_b) sums around the control volume loop in terms of the desired mass flow rate parameter, the final form of the pressure loop balance equation is obtained by substituting Equations (6.6) and (6.15) into Equation (6.1) as follows

$$\left(\frac{\sum k}{2\rho_m A_c^2}\right)\dot{m}^2 + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)\dot{m} - gL\rho_o \left(\frac{QS}{\dot{m}C_{Pm}T_m}\right) = 0 \tag{6.16}$$

and, multiplying Equation (6.16) by the mass flow rate ($\dot{m}$) yields

$$\left(\frac{\sum k}{2\rho_m A_c^2}\right)\dot{m}^3 + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)\dot{m}^2 - gL\rho_o \left(\frac{QS}{C_{Pm}T_m}\right) = 0 \tag{6.17}$$

Equation (6.17) represents the final expression for the mass flow rate solution at steady-state conditions, which satisfies the problem description provided in Figure 6.1 above. Combined with a second equation from the enforcement of energy conservation for the mean air temperature (T_m), derived from Equation (6.14) as

$$T_m = \frac{QS}{\dot{m}C_{Pm}} + T_{in} \tag{6.18}$$

A complete set of governing equations necessary for obtaining a final solution to this simple example problem by a chosen method of iteration is given by Equations (6.17) and (6.18). The exact solution method employed within the current program version is discussed in Section 6.3 below with the development of program code.

However, prior to the disclosure of a specific solution method, the relevant theory and governing equations developed through consideration of the simple example provided above warrant a brief demonstration on the extension of these formulations toward the characterization of actual flow path segments defined in the final single loop analysis of the detailed fuel assembly control volume. An illustration of these flow path segments integrated into program code calculations along with a description of the single loop analysis setup and nomenclature are provided in Figure 6.3 below. As mentioned in Section 5.3 and further discussed in Section 6.4, the selected number of discrete flow path segments is indicative of the highest resolution potential that SNL is capable of retaining for calibrating the MELCOR SFP model. This stems from the limited decomposition of domain data, which is inherently defined by the number and location of pressure ports utilized in SNL experiments for measuring the pressure losses.

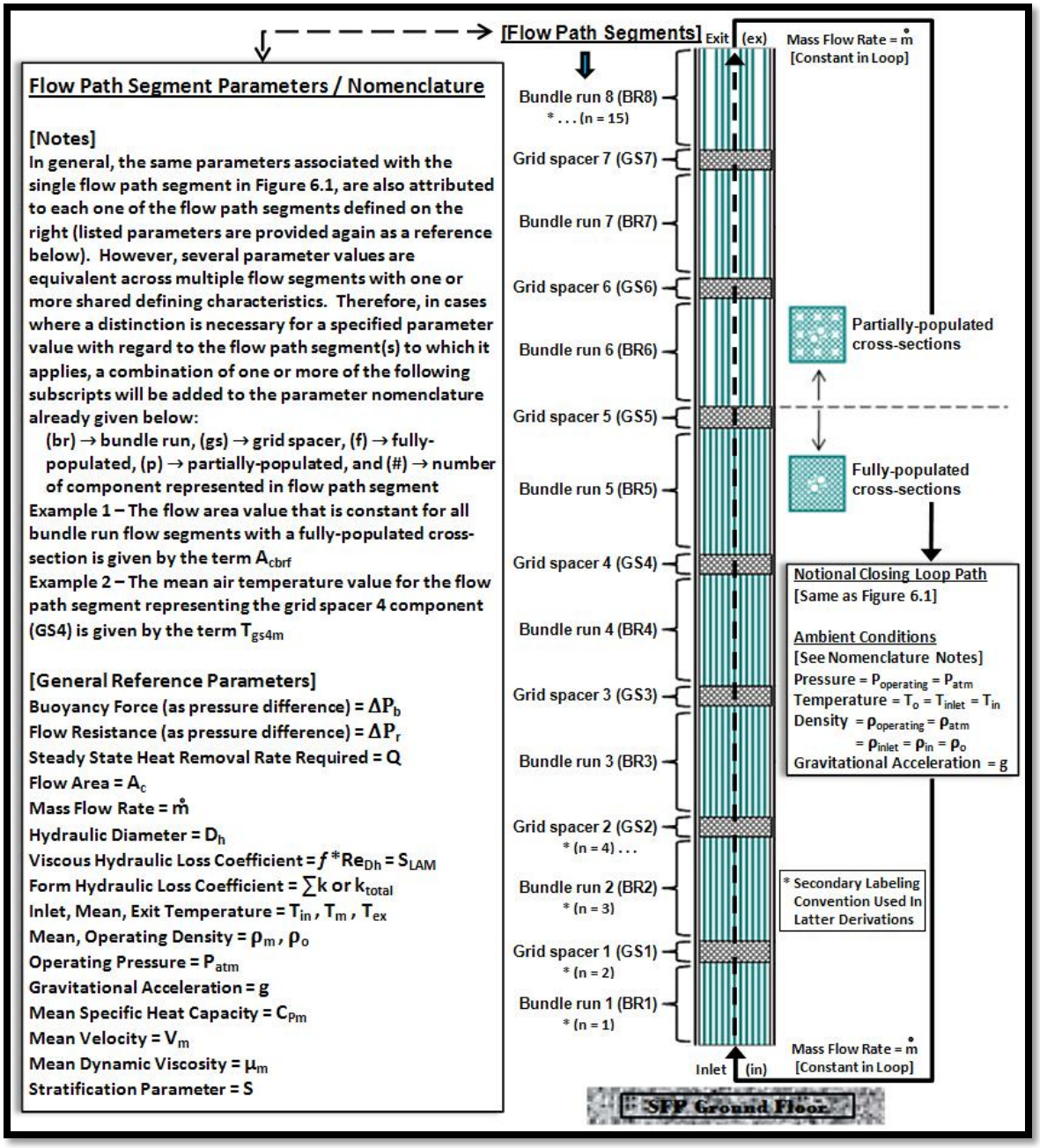


Figure 6.3: Detailed representation of actual flow path segments considered in the MATLAB buoyancy-driven flow program calculations accompanied by general parameter definitions and summary of nomenclature convention.

The single loop analysis of the actual assembly control volume shown in Figure 6.3 involves calculations for a significant number of flow path segments, which is markedly different in comparison to the single flow path segment of the initial example defined in Figure 6.1 above. However, in essence, derivation of the governing equations for a control volume loop encompassed by multiple flow path segments is simply an extension of the relevant theory and thought process implemented for the single flow path example, but applied to each flow path segment individually in consideration of the final governing equations of the control volume loop as a whole. In other words, the same essential form of the governing equations derived for the single flow path segment of the aforementioned example are restated in terms of each flow path segment illustrated in Figure 6.3. These expressions are subsequently combined to obtain a final set of equations representing the identical physical phenomena captured in the simplified example, but at a substantially greater resolution of the solution domain.

Therefore, by following the method of development for Equation (6.6) and exercising appropriate parameter definitions/nomenclature defined in Figure 6.3, a representative equation is obtained for the imposed resistance to flow in each individual flow path segment. Combining the expressions from all flow path segments for the imposed resistance to flow and grouping terms by order of the mass flow rate parameter, the resulting statement for the total resistance to flow around the control volume loop (ΔP_r) in abbreviated form is

$$\Delta P_r = \left[\left(\frac{\sum k}{2\rho_m A_c^2}\right)_{br1} + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{gs1} + \cdots\cdots + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{br8}\right]\dot{m}^2 + \left[\left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{br1} + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{gs1} + \cdots\cdots + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{br8}\right]\dot{m} \tag{6.19}$$

At the outset of consideration, the driving buoyancy pressure differential around the control volume loop is determined in a nearly identical manner to the total resistance to flow in which the total driving pressure differential (ΔP_b) is simply the sum of all buoyancy forces across the individual flow path segments. The general equation for the buoyancy force in terms of a pressure differential is given by the form of Equation (6.10) for each flow path segment as

$$\Delta P_{bpath} = g L_{path} \rho_o \left[\frac{(T_m - T_{in})}{T_m} \right]_{path} \tag{6.20}$$

However, the expression for the mean temperature difference with ambient external air in Equation (6.20) is unique for an individual segment due to the representative statement for the stratification parameter and energy balance associated with each flow path. Therefore, a more in-depth derivation is required in order to obtain the appropriate formulations substituted into Equation (6.20) for all flow path segments present in the control volume loop.

As an example, consider Equations (6.11) and (6.12) representing the energy balance and stratification parameter definitions for a single flow path segment, respectively. Explicitly writing these equations for the flow path segment (BR1) in Figure 6.3 above, affords statements for the heat rate balance and stratification parameter accordingly as follows

$$Q_{br1} = \dot{m} C_{Pbr1m} (T_{br1ex} - T_{in}) \tag{6.21}$$

$$T_{br1m} = T_{in} + S_{br1}(T_{br1ex} - T_{in}) \tag{6.22}$$

Rearranging Equation (6.22) for the temperature differential $(T_{br1ex} - T_{in})$ and substituting the result into Equation (6.21), yields

$$Q_{br1} = \dot{m} C_{Pbr1m} \left[\frac{(T_{br1m} - T_{in})}{S_{br1}} \right] \tag{6.23}$$

Solving Equation (6.23) for the mean temperature difference with ambient external air $(T_{br1m} - T_{in})$, gives

$$(T_{br1m} - T_{in}) = \frac{Q_{br1} S_{br1}}{\dot{m} C_{Pbr1m}} \tag{6.24}$$

Substituting Equation (6.24) into Equation (6.20), the desired form for the buoyancy pressure differential across flow path segment (BR1) becomes

$$\Delta P_{bbr1} = \frac{g L_{br1} \rho_o}{T_{br1m}} \left(\frac{Q_{br1} S_{br1}}{\dot{m} C_{Pbr1m}} \right) \tag{6.25}$$

and, using Equation (6.24) again, a second necessary equation for the mean air temperature (T_{br1m}) is obtained for the enforcement of energy conservation as

$$T_{br1m} = \left(\frac{Q_{br1} S_{br1}}{\dot{m} C_{Pbr1m}} \right) + T_{in} \tag{6.26}$$

Equations (6.25) and (6.26) complete the required set of equations for (BR1), and the procedure is repeated for each flow path segment in Figure 6.3 in order to construct the final set of governing equations implemented into the program code.

Again, as further demonstration of the increased complexity associated with each subsequent flow path segment, consider the equations required for the ensuing (GS1) segment immediately downstream of (BR1). As with the remaining flow path segments in Figure 6.3, the primary objective is to derive an equivalent relationship for the mean temperature difference with ambient external air ($T_m - T_{in}$) in terms of the mass flow rate parameter, as given by Equation (6.24) for segment (BR1) above. Hence, the initial heat rate balance and stratification parameter for (GS1) are respectively written

$$Q_{gs1} = \dot{m} C_{Pgs1m} \left(T_{gs1ex} - T_{br1ex} \right) \tag{6.27}$$

$$T_{gs1m} = T_{br1ex} + S_{gs1} \left(T_{gs1ex} - T_{br1ex} \right) \tag{6.28}$$

Rearranging Equation (6.28) for the temperature differential ($T_{gs1ex} - T_{br1ex}$) and substituting into Equation (6.27), yields

$$Q_{gs1} = \dot{m} C_{Pgs1m} \left[\frac{\left(T_{gs1m} - T_{br1ex} \right)}{S_{gs1}} \right] \tag{6.29}$$

or, in a slightly different form

$$\frac{Q_{gs1} S_{gs1}}{\dot{m} C_{Pgs1m}} = \left(T_{gs1m} - T_{br1ex} \right) \tag{6.30}$$

Noting again the objective to acquire an equivalent relationship for the mean temperature difference with ambient external air ($T_{gs1m} - T_{in}$) in terms of the mass flow rate parameter, an expression for (T_{br1ex}) in Equation (6.30) is obtained from Equation (6.22) as

$$T_{br1ex} = \frac{(T_{br1m} - T_{in})}{S_{br1}} + T_{in} \tag{6.31}$$

or, by using Equation (6.24) to rewrite the first group of terms on the right-hand side of Equation (6.31), (T_{br1ex}) is also given by

$$T_{br1ex} = \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} + T_{in} \tag{6.32}$$

Substituting Equation (6.32) into Equation (6.30), yields

$$\frac{Q_{gs1}S_{gs1}}{\dot{m}C_{Pgs1m}} = T_{gs1m} - T_{in} - \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} \tag{6.33}$$

Thus, by rearranging Equation (6.33), an equation for the mean temperature difference with ambient external air ($T_{gs1m} - T_{in}$) in flow path segment (GS1) becomes

$$\left(T_{gs1m} - T_{in}\right) = \frac{Q_{gs1}S_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} \tag{6.34}$$

Substituting Equation (6.34) into Equation (6.20), the desired form for the buoyancy pressure differential across (GS1) is

$$\Delta P_{bgs1} = \frac{gL_{gs1}\rho_o}{T_{gs1m}}\left(\frac{Q_{gs1}S_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}}\right) \tag{6.35}$$

and, using Equation (6.34) again, an equation for the mean air temperature (T_{gs1m}) is obtained for the enforcement of energy conservation as

$$T_{gs1m} = \frac{Q_{gs1}S_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} + T_{in} \tag{6.36}$$

Equations (6.35) and (6.36) complete the required set of equations for (GS1), and the procedure is continued with (BR2) and each subsequent flow path segment in Figure 6.3 above.

Although these derivations become increasingly tedious to write out, a pattern develops for establishing these equations with each progressive flow path segment in the downstream direction. This pattern becomes apparent after deriving the expressions for mean temperature and mean temperature difference with ambient external air in the ensuing (BR2) flow path segment. However, for brevity, only the final results of this derivation are provided below. Hence, by pursuing a similar procedure as outlined for (GS1), statements are obtained for the mean temperature difference with ambient external air and mean temperature in flow path segment (BR2) respectively as follows

$$(T_{br2m} - T_{in}) = \frac{Q_{br2}S_{br2}}{\dot{m}C_{Pbr2m}} + \frac{Q_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} \tag{6.37}$$

$$T_{br2m} = \frac{Q_{br2}S_{br2}}{\dot{m}C_{Pbr2m}} + \frac{Q_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} + T_{in} \tag{6.38}$$

and, substituting Equation (6.37) into Equation (6.20), the desired form for the buoyancy pressure differential across (BR2) becomes

$$\Delta P_{bbr2} = \frac{gL_{br2}\rho_o}{T_{br2m}}\left(\frac{Q_{br2}S_{br2}}{\dot{m}C_{Pbr2m}} + \frac{Q_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}}\right) \tag{6.39}$$

Equations (6.38) and (6.39) are the complete set of expressions necessary for the (BR2) flow path segment.

At this point, examination of Equations (6.25), (6.35), and (6.39) reveals an obvious pattern development between the statements of buoyancy pressure differential (ΔP_{bpath}) for each progressive flow path segment in the downstream direction. Expressing this pattern in terms of a general equation for all flow path segments is best achieved by applying a new labeling mechanism whereby each path segment in Figure 6.3 is successively numbered in the downstream direction from (n = 1) for (BR1) to (n = 15) for (BR8). Using this nomenclature convention, a generalized expression for the buoyancy pressure differential ($\Delta P_{b(n)}$) across any individual flow path segment (n) is given by

$$\Delta P_{b(n)} = \frac{gL_{(n)}\rho_o}{\dot{m}T_{(n)m}}\left[\frac{Q_{(n)}S_{(n)}}{C_{P(n)m}}\left[\left(+\sum_{i=1}^{(n-1)}\frac{Q_{(i)}}{C_{P(i)m}}\right)\{for\ n > 1\}\right]\right] \tag{6.40}$$

For example, employing Equation (6.40) to obtain the buoyancy pressure differential for flow path segment (GS2) designated by the label (n = 4), yields

$$\Delta P_{b(4)} = \frac{gL_{(4)}\rho_o}{\dot{m}T_{(4)m}}\left(\frac{Q_{(4)}S_{(4)}}{C_{P(4)m}} + \frac{Q_{(3)}}{C_{P(3)m}} + \frac{Q_{(2)}}{C_{P(2)m}} + \frac{Q_{(1)}}{C_{P(1)m}}\right) \tag{6.41}$$

According to the prescribed labeling convention, Equation (6.41) is equivalently expressed as

$$\Delta P_{bgs2} = \frac{gL_{gs2}\rho_o}{\dot{m}T_{gs2m}}\left(\frac{Q_{gs2}S_{gs2}}{C_{Pgs2m}} + \frac{Q_{br2}}{C_{Pbr2m}} + \frac{Q_{gs1}}{C_{Pgs1m}} + \frac{Q_{br1}}{C_{Pbr1m}}\right) \tag{6.42}$$

Therefore, the generalized expression given by Equation (6.40) in conjunction with the new labeling mechanism, permits casting of the total buoyancy pressure differential around the entire control volume loop in an abbreviated form as follows

$$\Delta P_b = \frac{g\rho_o}{\dot{m}}\left[T_{delta(n=1)} + T_{delta(n=2)} + \cdots\cdots + T_{delta(n=15)}\right] \tag{6.43}$$

where, using a modified representation of the derivation for Equation (6.40) above, the ($T_{delta(n)}$) terms in Equation (6.43) are given by

$$T_{delta(n)} = \frac{L_{(n)}}{T_{(n)m}}\left[\frac{Q_{(n)}S_{(n)}}{C_{P(n)m}}\left[\left(+\sum_{i=1}^{(n-1)}\frac{Q_{(i)}}{C_{P(i)m}}\right)\{for\ n > 1\}\right]\right] \tag{6.44}$$

Equation (6.43) and the parameter definition from Equation (6.44), provide the concluding statements necessary for adequate description of the buoyancy pressure differential (ΔP_b) around the control volume loop.

Predicated on successful derivation of expressions for (ΔP_r) and (ΔP_b), the final governing equation characterizing the overall balance of total resistance to flow and

driving buoyancy pressure differential around the control volume loop is acquired in abbreviated form by substituting Equations (6.19) and (6.43) into Equation (6.1). However, prior to substitution, Equation (6.19) is re-expressed in accordance with the new labeling convention to maintain a congruent nomenclature with remaining terms. Thus, the governing equation for the overall pressure balance in the single loop analysis is written as

$$\left[\left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=1)} + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=2)} + \cdots\cdots + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=15)}\right]\dot{m}^2 + \left[\left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=1)} + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=2)} + \cdots\cdots + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=15)}\right]\dot{m} - \left[\left(\frac{g\rho_o}{\dot{m}}\right)\left[T_{delta(n=1)} + T_{delta(n=2)} + \cdots\cdots + T_{delta(n=15)}\right]\right] = 0 \tag{6.45}$$

Multiplying through by the mass flow rate ($\dot{m}$), Equation (6.45) becomes

$$\left[\left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=1)} + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=2)} + \cdots\cdots + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=15)}\right]\dot{m}^3 + \left[\left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=1)} + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=2)} + \cdots\cdots + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=15)}\right]\dot{m}^2 - \left[(g\rho_o)\left[T_{delta(n=1)} + T_{delta(n=2)} + \cdots\cdots + T_{delta(n=15)}\right]\right] = 0 \tag{6.46}$$

where, ($T_{delta(n)}$) in Equation (6.46) is given by Equation (6.44), which is written again for reference as follows

$$T_{delta(n)} = \frac{L_{(n)}}{T_{(n)m}} \left[\frac{Q_{(n)} S_{(n)}}{C_{P(n)m}} \left[\left(+ \sum_{i=1}^{(n-1)} \frac{Q_{(i)}}{C_{P(i)m}} \right) \{for\ n > 1\} \right] \right]$$

Equation (6.46) with parameter specifications determined by Equation (6.44), collectively constitute the final governing equation for the force/pressure balance around the control volume loop in Figure 6.3, which is subsequently utilized in solving for the mass flow rate solution at steady-state conditions.

On a final note, an expression for the mean temperature in each flow path segment stemming from the heat rate balance, written in terms of the mass flow rate, is required as a method of enforcing the conservation of energy across each individual subvolume component from the fuel assembly inlet to outlet. This set of formulas embodies the remaining requisite governing equations for the single loop analysis calculations implemented into the MATLAB program code. Obviously, since the mean temperature is explicitly included in the statement of mean temperature difference with ambient external air, which is conveyed in the primary pattern associated with the generalized expression for ($\Delta P_{b(n)}$) in Equation (6.40), a similar pattern also exists for the mean temperature relationship of progressive flow path segments.

The pattern development between statements of mean temperature for successive flow path segments is revealed through examination of Equations (6.26), (6.36), and (6.38) above. Continuing with the new labeling convention, a generalized expression for the mean temperature ($T_{(n)m}$) of any individual flow path segment (n) is given by

$$T_{(n)m} = \frac{Q_{(n)} S_{(n)}}{\dot{m} C_{P(n)m}} \left[\left(+ \sum_{i=1}^{(n-1)} \frac{Q_{(i)}}{\dot{m} C_{P(i)m}} \right) \{for\ n > 1\} \right] + T_{in} \tag{6.47}$$

For example, the mean temperature of flow path segment (GS2) designated by the label ($n = 4$) is obtained using Equation (6.47) as follows

$$T_{(4)m} = \frac{Q_{(4)}S_{(4)}}{\dot{m}C_{P(4)m}} + \frac{Q_{(3)}}{\dot{m}C_{P(3)m}} + \frac{Q_{(2)}}{\dot{m}C_{P(2)m}} + \frac{Q_{(1)}}{\dot{m}C_{P(1)m}} + T_{in} \tag{6.48}$$

According to the prescribed labeling convention, Equation (6.48) is equivalently expressed as

$$T_{gs2m} = \frac{Q_{gs2}S_{gs2}}{\dot{m}C_{Pgs2m}} + \frac{Q_{br2}}{\dot{m}C_{Pbr2m}} + \frac{Q_{gs1}}{\dot{m}C_{Pgs1m}} + \frac{Q_{br1}}{\dot{m}C_{Pbr1m}} + T_{in} \tag{6.49}$$

Thus, Equation (6.47) affords a simple, generalized expression for individually obtaining the mean temperatures of all flow path segments in Figure 6.3, which is required to ensure the conservation of energy across each flow path segment.

In conclusion, a complete set of the final governing equations implemented into the MATLAB program code calculations is comprised of Equation (6.46) with parameter specifications determined by Equation (6.44) and Equation (6.47) written out for the mean temperature of every flow path segment. The relevant theory and governing equations developed through consideration of the examples outlined above provide sufficient introduction and description necessary for a complete understanding of the program code development and numerical solution procedure addressed in the ensuing discussions.

6.3 DEVELOPMENT OF PROGRAM CODE

The principal focus of initial development efforts for any computational analysis program, after formulating a basic set of governing equations, is the establishment of an adequate solution method that inherently provides the skeletal structure of the program code. Consequently, the following discussions aim to introduce the solution procedure incorporated into the desired numerical analysis, describe the necessary programming logic through the outline of an appropriate pseudocode, and present a brief review of the final key sections of applied code and essential program functions. A firm understanding of the exact governing equations under consideration is critical in detailing the specific solution method and a restatement of these expressions is warranted below as a quick reference to the succeeding deliberation. Once more, the final set of governing equations derived in Section 6.2 consists of Equation (6.46) with defined parameters from Equation (6.44) as well as the formulation of mean temperature for each flow path segment from Equation (6.47), which are respectively depicted again as follows:

$$\left[\left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=1)} + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=2)} + \cdots\cdots + \left(\frac{\sum k}{2\rho_m A_c^2}\right)_{(n=15)}\right]\dot{m}^3$$
$$+\left[\left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=1)} + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=2)} + \cdots\cdots + \left(\frac{S_{LAM}\mu_m L}{2D_h^2\rho_m A_c}\right)_{(n=15)}\right]\dot{m}^2$$
$$-\left[(g\rho_o)\left[T_{delta(n=1)} + T_{delta(n=2)} + \cdots\cdots + T_{delta(n=15)}\right]\right] = 0$$

where,

$$T_{delta(n)} = \frac{L_{(n)}}{T_{(n)m}}\left[\frac{Q_{(n)}S_{(n)}}{C_{P(n)m}}\left[\left(+\sum_{i=1}^{(n-1)}\frac{Q_{(i)}}{C_{P(i)m}}\right)\{for\ n > 1\}\right]\right]$$

and

$$T_{(n)m} = \frac{Q_{(n)}S_{(n)}}{\dot{m}C_{P(n)m}}\left[\left(+\sum_{i=1}^{(n-1)}\frac{Q_{(i)}}{\dot{m}C_{P(i)m}}\right)\{for\ n>1\}\right] + T_{in}$$

A successful iterative solution method is stipulated by the capacity to facilitate accurate solutions to a range of computational analyses under varying, but acceptable limits of initial conditions with convergence attained in a reasonably efficient and routine manner. Thus, a property of particular importance in a desired solution method is the progressively improved estimation of unknown parameter values within a predetermined number of iterations. Examining the governing equations above and disregarding parameters with known (and presumed known) values, reveals the following clarifications toward the development of a solution procedure: (1) primary unknowns include mass flow rate and mean temperatures as well as mean temperature dependent air properties in flow path segments; (2) explicit influence of mean temperatures in the pressure balance equation is limited to the buoyancy pressure differential terms; and (3) total resistance to flow is only indirectly influenced by mean temperatures via the dependence of air property values. Furthermore, mass flow rate in the control volume loop is correlated with mean temperatures throughout the flow domain by the governing equation associated with each flow path segment for the mean temperature.

As a result, the mass flow rate parameter affords a critical interlink necessary in order to evaluate the correctness of mean temperature values for individual flow path segments and successively provide a better estimate of these temperatures based on results from the previous iteration. Generally, an initial solution method proposed for

solving any numerical problem embodies the most straightforward calculations possible that demand the fewest number of steps to obtain an accurate solution. Supplied with initial guesstimates of the mean temperature in all flow path segments and the corresponding air property values, an initial iterative solution process proposed for resolving the governing equations above is likely to involve a few fundamental steps: (1) solving for the mass flow rate [Equations (6.46) and (6.44)]; (2) substituting the result into mean temperature equations of individual flow path segments and calculating the mean temperature that satisfies the heat rate balance in each [Equation (6.47)]; (3) updating the mean temperature estimates based on current and prior iteration results (*i.e.* the average value) if convergence criterion is not met in all flow path segments; (4) calculating air properties at the new mean temperature estimates; and (5) repeating the above process until convergence is achieved. However, employing a solution method in which the mean temperature and dependent air property values are concurrently updated in every iteration cycle is problematic at best. The two inherent difficulties associated with the initially proposed method are thoroughly explained below.

1. Although solutions from Equation (6.47) provide better estimates of the final mean temperatures in the present domain, these values are not necessarily better estimates of the final mean temperatures in the domain characterized by air properties at the new mean temperature estimates. In other words, the values determined by Equation (6.47) are only presumed as better estimates of the final mean temperatures in the domain of currently prescribed air properties. Thus, a solution method in which the mean temperature and dependent air property values are concurrently updated in every iteration cycle is likely to exhibit

unbounded/divergent behavior in certain computational analyses and more often fail to reach a converged solution.

2. At the end of each iteration cycle, current and prior iteration results from Equation (6.47) for the mean temperatures of flow path segments are used to establish better estimates of these values for the next iteration. The mean temperature differential between successive iteration estimates in each individual flow path segment undoubtedly influence the following with different magnitudes: (1) the net change of driving buoyancy pressure through direct modification of mean temperature estimates and (2) the net change of both driving buoyancy and resistance to flow pressures through indirect modification of the mean temperature dependent air properties.

   In other words, the magnitude of change effected in the mass flow rate solution by mean temperature and air property influences, stemming from perturbed mean temperature estimates, are unequal. Hence, concurrently updating mean temperature and dependent air property values inevitably results in the dampening of either mean temperature or air property influence on the iterative solution process by the other, obviously more-dominant term(s) (susceptible to exchange during a computation). Furthermore, the iterative solution process is likely to become cyclic due to the unbalanced influence of unknown terms on mass flow rate and mean temperature estimates.

Therefore, a systematic approach consisting of separate sub-calculations is necessary within the solution method in order to control/steer the iteration process toward convergence. Fortunately, a practical resolution exists based simply on the three clarification points described above for the development of a general solution procedure.

Due to the direct influence of mean temperature estimates arising only in the group of buoyancy pressure differential terms for the pressure loop balance [Equation (6.46)], the effects of successive iteration upon mean temperature estimates for the flow path segments and correlated changes in air properties are easily separated into distinct iteration sub-loops for independent consideration throughout the solution process. This systematic approach permits control over the iteration cycles by progressively calculating improved mean temperature estimates until a specified criterion is met followed by a delineated outer calculation loop that subsequently updates the air property values in a repeated fashion. Hence, through the addition of a few intermediate sub-calculations, the initially proposed solution process is dramatically improved by incorporating systematic control steps into the iterative procedure as summarized below.

1. Supplied with initial guesstimates of the mean temperature in all flow path segments and the corresponding air property values, solve for the mass flow rate [Equation (6.46) with Equation (6.44)].
2. Substitute result into mean temperature equations of individual flow path segments and calculate mean temperature that satisfies heat rate balance in each [Equation (6.47)].
3. If convergence criterion is not met in all flow path segments, update only the mean temperature estimates based on current and prior iteration results (*i.e.* the average value).
4. Repeat the above steps until convergence criterion for the mean temperature in all flow path segments is attained for the air properties evaluated at initial mean temperature guesstimates.

5. Compare initial mean temperature guesstimates with the current converged mean temperatures to check for convergence of air property values; based on differential between mean temperatures used to determine air properties in prior iteration cycle to newly converged mean temperatures in flow path segments.
6. If convergence criterion is not met in all flow path segments, update air properties in all segments at newly converged mean temperatures and repeat the procedure above until convergence is achieved for the air property evaluations of all flow path segments.

The solution methodology described by the logic above provides the platform necessary in order to formulate the initial structure of the program code. Founded on a thorough consideration of the calculations outlined in each of these steps, the entire code is constructed as a compilation of three individual MATLAB M-files including: (1) a script M-file serving as the spine of the code for general input/output as well as control of program calculations and function calls, called buoyancy.m; (2) a function M-file that calculates mass flow rate solution(s) of Equation (6.46) using built-in MATLAB function roots(p) to find zeros of the polynomial equation, called massflowbuoyancy.m; and (3) a function M-file that calculates air property values at specified temperatures using curve fits established in Section 5.3 for FLUENT CFD simulations, called propertybuoyancy.m. Therefore, based on the solution method with general description of embodied M-files presented above, a pseudocode representation of the entire program code is developed in a straightforward manner as briefly outlined in Figure 6.4 below.

BRIEF OUTLINE OF PSEUDO-CODE DEVELOPMENT FOR MATLAB PROGRAM

START PROGRAM – INITIALIZE SCRIPT M-FILE BUOYANCY.M
Clear memory and declare global variables
Display program description and general information
Define or prompt user for required parameter inputs and initial estimates
Call function M-file propertybuoyancy.m for initial air properties at starting mean temperatures
Initialize starting values for outer and inner loop calculations

•BEGIN OUTER LOOP FOR UPDATING AIR PROPERTIES
Initialize or update count for max number of outer loop iterations

•BEGIN INNER LOOP FOR APPROXIMATING MEAN TEMPERATURES
Initialize or update count for max number of inner loop iterations
Call function M-file massflowbuoyancy.m for mass flow rate solution(s) of Equation (6.46)
Calculate mean temperatures for heat rate balances of flow path segments from Equation (6.47)
Check if convergence criterion is satisfied for all flow path segments
If yes, break inner loop
If no, update to better estimates of mean temperatures based on current and prior iteration results
Return to beginning of inner loop and repeat until convergence or max number of iterations reached
•END INNER LOOP

Display intermediate converged results from inner loop approximations of mean temperatures
Compare difference in mean temperatures used to last evaluate air properties with new inner loop results
Check if convergence criterion is satisfied for air properties in all flow path segments
If yes, break outer loop
If no, call function M-file propertybuoyancy.m and evaluate air properties at new mean temperatures
Store new mean temperature values used to evaluate air properties for comparison with next cycle results
Return to beginning of outer loop and repeat until convergence or max number of iterations reached
Also, use converged inner loop mean temperatures as new inner loop initial mean temperature estimates
•END OUTER LOOP

Calculate and display final results from current model run
STOP PROGRAM

Figure 6.4: Brief outline of pseudocode development for MATLAB buoyancy-driven flow program.

A complete copy of the final program code is provided in Appendix A, which includes an example model run with the entire command window input/output for reference. In addition, an extensive scope of comments accompanies the final program copy in Appendix A affording sufficient description of nearly every line of code and clear-cut comprehension of the computational tasks corresponding to each distinct

segment of the code. Exceeding one thousand numbered lines of code, the substantial size of the program precludes any form of practical depiction or necessity to elaborate upon the description of program code any further in the present discussion, especially outside sufficiently detailed comments already included in Appendix A for nearly every line of code. In other words, attention is directed toward a brief review of the final program code and comments presented in Appendix A in lieu of a redundant, superfluous consideration of the individual lines of code in the current discourse. The ensuing discussions with regard to the program validation and critical heat input rate studies provide additional insight and reference into a number of principal segments of the program code as well.

6.4 Program Validation and Critical Heat Input Rate Studies

Establishing the validity of a computational analysis program is of utmost importance in the concluding stages of any development effort, which requires demonstrating the capability to obtain accurate solutions through comparisons with benchmark results, ensuring a level of confidence in solutions through consistency of program responses to perturbations, and robustness of the program application through parameter sensitivity studies. The purpose of the following research efforts is to address these concerns by comparing the MATLAB program results with solutions obtained from the FLUENT CFD heated simulations in Section 5.3.3, investigating the sensitivity of required input values on predicted solution responses by the program, and determining critical heat input rates utilizing both computational and SNL experimental results for comparative analysis. In addition, a unique method for approximating the viscous

hydraulic loss coefficient (S_{LAM}) associated with multipath cross-sections of flow is introduced and briefly evaluated for consideration in a few numerical trials.

6.4.1 Comparison with FLUENT CFD Simulations

The most important and crucial task undertaken in establishing the validity of the MATLAB computational analysis program involves a thorough benchmarking of the results against FLUENT CFD simulations for the heated buoyancy-driven flow trials in Section 5.3.3 using the equivalent fuel assembly model under congruent setup conditions. As previously mentioned, the boundary conditions and parameter input values implemented into the MATLAB program analyses conducted below are identical to those enforced in the FLUENT CFD simulations (where applicable) unless otherwise noted. Several of the results from these CFD trials in Section 5.3.3 are required as inputs to the MATLAB program for preliminary comparison including the viscous and form hydraulic loss coefficients determined from curve fits to the pressure loss data.

Although an extent of the results is already presented throughout Section 5.3, the solution data is included again in Tables 6.1 and 6.2 below in accompaniment with essential program inputs and initial estimates utilized in the MATLAB computations performed. The range of values examined in these comparative trials spans from 300 to 1200 watts for the net decay heat input rate of the fuel.

Table 6.1: Reference input values for parameters and initial estimates required by MATLAB program in base comparison trials performed (1 of 2).

MATLAB PROGRAM - BASE COMPARISON TRIALS - REFERENCE INPUT VALUES: KNOWN PARAMETERS AND INITIAL ESTIMATES

[Geometry]	Value:
Bundle Run Length (m)	0.473330
Grid Spacer Length (m)	0.030480
Total Model Length (m)	4.000000
Fully-Populated Bundle Flow Area (m^2)	9.688292E-03
Fully-Populated Spacer Flow Area (m^2)	7.267489E-03
Partially-Populated Bundle Flow Area (m^2)	1.046384E-02
Partially-Populated Spacer Flow Area (m^2)	8.251381E-03

[Geometry - Continued]	Value:
Fully-Populated Bundle Hydraulic Diameter (m)	0.01190721
Fully-Populated Spacer Hydraulic Diameter (m)	0.00290797
Partially-Populated Bundle Hydraulic Diameter (m)	0.01406727
Partially-Populated Spacer Hydraulic Diameter (m)	0.00359172

[Ambient Environment Conditions]	Value:
Pressure (Pa)	82833.7622
Temperature (K)	294.4034
Gravitational Acceleration (m/s^2)	9.81

Hydraulic Loss Coefficients - Determined By CFD Heated Simulations Using 'Equivalent Fuel Assembly Model' From (300 - 1200 Watts)

[Viscous Terms - S_{LAM}]	Value:
Fully-Populated Inlet Bundle Run ()	109.174754
Fully-Populated Bundle Run ()	98.808304
Fully-Populated Grid Spacer ()	66.241816
Partially-Populated Bundle Run ()	98.035538
Partially-Populated Grid Spacer ()	43.359972

[Form Terms - k]	Value:
Fully-Populated Inlet Bundle Run ()	-4.268716
Fully-Populated Bundle Run ()	0.300386
Fully-Populated Grid Spacer ()	0.128565
Partially-Populated Bundle Run ()	0.930330
Partially-Populated Grid Spacer ()	0.318634

[Additional Inlet Loss Term - f_1]	Value:
Enhancement Due To Flow Development ()	0.96

Note: Also termed the pressure defect number, this parameter accounts for additional losses (typically minute) across the inlet bundle run due to initial flow development/initial setting of fluid in motion from expansion of air. Value is based on conservative estimate from published experimental results for heated flows in high aspect ratio ducts in the literature (Brinkworth *et al.* , 2000).

Table 6.2: Reference input values for parameters and initial estimates required by MATLAB program in base comparison trials performed (2 of 2).

[Stratification Parameters - S]

Note: Stratification parameter values for individual flow path segments are dependent upon the net decay heat input rate and determined by default, built-in functions. These functions are based on curve-fits to the stratification versus net heat input rate value data from heated CFD simulations using 'equivalent fuel assembly model' for each of the flow path segments from (300 - 1200 Watts).

[Heat Input Rates - Q]	Value:	
Net Decay Heat Input Rate (W)	300 - 1200	(50 Watt Increments)

Note: Heat input rates for individual flow path segments are based on a fraction of the total length of all fuel rods in a flow segment to the total length of all fuel rods in the entire fuel assembly multiplied by the net decay heat input rate.

[Initial Mean Temperature Estimates - T_m]

Average Flow Path Segment Value From 300 - 1200 Watts

Mean Temperature (K) - Bundle Runs	BR1:	BR2:	BR3:	BR4:	BR5:	BR6:	BR7:	BR8:
	332.79	417.72	500.69	581.63	660.74	734.56	802.64	869.63
Mean Temperature (K) - Grid Spacers	GS1:	GS2:	GS3:	GS4:	GS5:	GS6:	GS7:	
	376.97	460.55	542.23	622.04	700.22	772.20	839.64	

Note: Initial mean temperature estimates for individual flow path segments (listed above) are the same across all net decay heat input rates examined. These values are based on an average of the final mean temperatures for each flow path segment in heated CFD simulations using 'equivalent fuel assembly model' from (300 - 1200 Watts).

[Initial Mass Flow Rate Estimate - m*]	Value:
Air Mass Flow Rate (kg/s)	0.00129512

Note: Initial mass flow rate estimate (listed above) is the same across all net decay heat input rates examined. This value is based on the average of resulting final mass flow rates in heated CFD simulations using 'equivalent fuel assembly model' from (300 - 1200 Watts).

[Solution Controls]	Value:	
Mean Temperature Convergence Criterion (K)	0.2	(Max change in mean temperature for all flow path segments between successive inner loop iterations)
Air Properties Convergence Criterion (K)	0.4	(Max change in evaluated mean temperature for air properties in all flow path segments between successive outer loop iterations)
Max Number Of Inner Loop Iterations ()	100	(Max number of iterations for estimating mean temperatures in flow path segments per evaluated set of air property values)
Max Number Of Outer Loop Iterations ()	100	(Max number of iterations for updating air properties at new mean temperature estimates in flow path segments)
Under-Relaxation Factor () - T_m Bundle Updates	0.5	(Under-relaxation factor for determining new mean temperature estimates in bundle run flow path segments)
Under-Relaxation Factor () - T_m Spacer Updates	0.5	(Under-relaxation factor for determining new mean temperature estimates in grid spacer flow path segments)

Despite the brief explanations provided for a number of input values in Tables 6.1 and 6.2 above, the importance of selecting/determining the references derived in part from FLUENT CFD solutions in Section 5.3.3 warrant additional clarification (excluding hydraulic loss coefficients, which are adequately detailed in prior discussions). In particular, the stratification parameter values for individual flow path segments are determined by default (optional), built-in functions based on the net heat input rate examined in the analysis. The function for each flow path segment is derived from a curve fit to the stratification versus net heat input rate data acquired in heated CFD simulation trials at specified rates of (300, 600, 900, and 1200 watts) outlined in Section 5.3.3 using the equivalent fuel assembly model.

An evaluation of the curve fits obtained by employing the TableCurve 2D software revealed negligible differences between the actual stratification parameter values and those calculated from the curve-fit functions for all flow path segments (ΔS_{max} < 0.07%). This optional default for selecting built-in functions to compute the stratification parameter values of flow path segments based on net heat input rate affords a substantial increase in the program efficiency, especially for conducting numerous experimental trials. The built-in stratification parameter functions are easily viewed in Appendix A, which includes a copy of the final program code containing the inherent default functions.

In order to alleviate concerns with regard to initial mean temperature estimates chosen for the individual flow path segments in every numerical analysis (each with a specific net heat input rate) over the evaluated range of (300 - 1200 watts), a single input value is assigned to each flow path segment as the initial mean temperature estimate for all computational trials. The value selected for each flow path segment is the average resulting mean temperature from the heated CFD simulation trials in Section 5.3.3 at net

heat input rates of (300, 600, 900, and 1200 watts) using the equivalent fuel assembly model. Furthermore, employing the constant set of values for initial mean temperature estimates over the numerous analyses with net heat input rates spanning (300 - 1200 watts) provides an assessment of the sensitivity between inputs for initial mean temperature in flow path segments and accuracy of the solution response. A built-in option also exists within the MATLAB program for automatically selecting these values as the default inputs for initial mean temperature in flow path segments. The average resulting mass flow rate from the heated CFD simulation trials in Section 5.3.3 at net heat input rates of (300, 600, 900, and 1200 watts) is similarly applied in defining the single input value assigned as the initial mass flow rate estimate for all computational trials.

A number of parameter results from the base comparison trials exploiting the MATLAB computational analysis program are identified for comparison with FLUENT CFD simulation solutions in Section 5.3.3 for the heated buoyancy-driven flow trials based on the potential for a meaningful comparative analysis toward the overall assessment of program validity and likelihood of drawing beneficial conclusions from these results. Accordingly, solution values from the heated CFD simulations and MATLAB program base trials for parameters of greatest significance are included in comparisons of mass flow rate versus net heat input rate, total pressure loss versus net heat input rate, and exit temperature versus net heat input rate. These results are illustrated below in Figures 6.5 through 6.7, respectively.

At a lesser extent, further assessment of program validity is manifested in several other noteworthy comparisons of mean parameter values per flow path segment over a range of net heat input rates including mean temperature, mean velocity (in bundle runs), and mean velocity (in spacers). These results are displayed in the succeeding graphs of Figures 6.8 through 6.10, respectively. Numerical values from preceding graphical

results for the most significant parameters are also tabulated and contrasted below in closing Tables 6.3 and 6.4 as a final reference for the comparison of solutions from heated CFD simulations in Section 5.3.3 and the MATLAB program base trials.

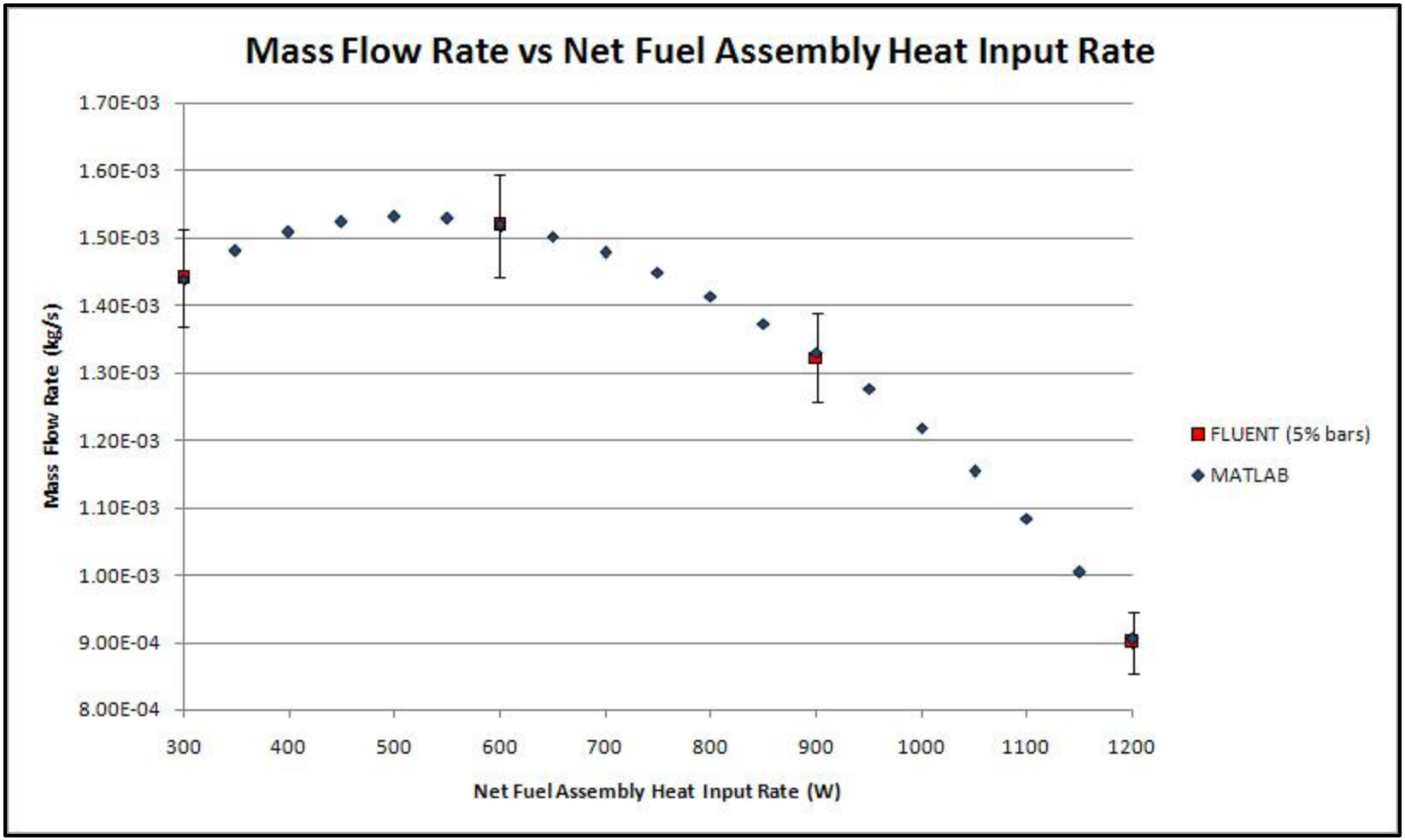


Figure 6.5: Comparison of MATLAB program base trials for mass flow rate versus net heat input rate with FLUENT CFD benchmark values determined from heated, buoyancy-driven flow simulations.

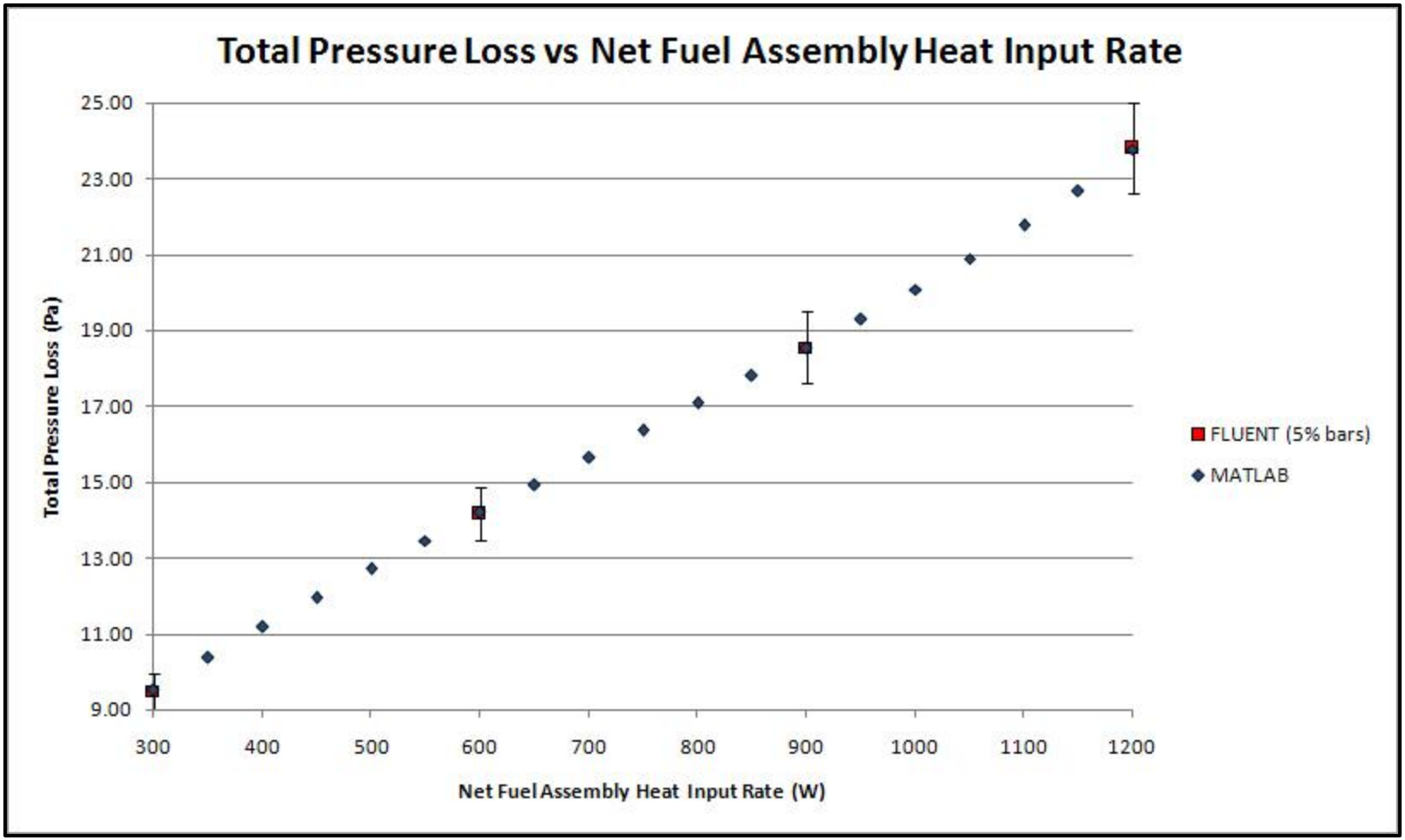


Figure 6.6: Comparison of MATLAB program base trials for total pressure loss versus net heat input rate with FLUENT CFD benchmark values determined from heated, buoyancy-driven flow simulations.

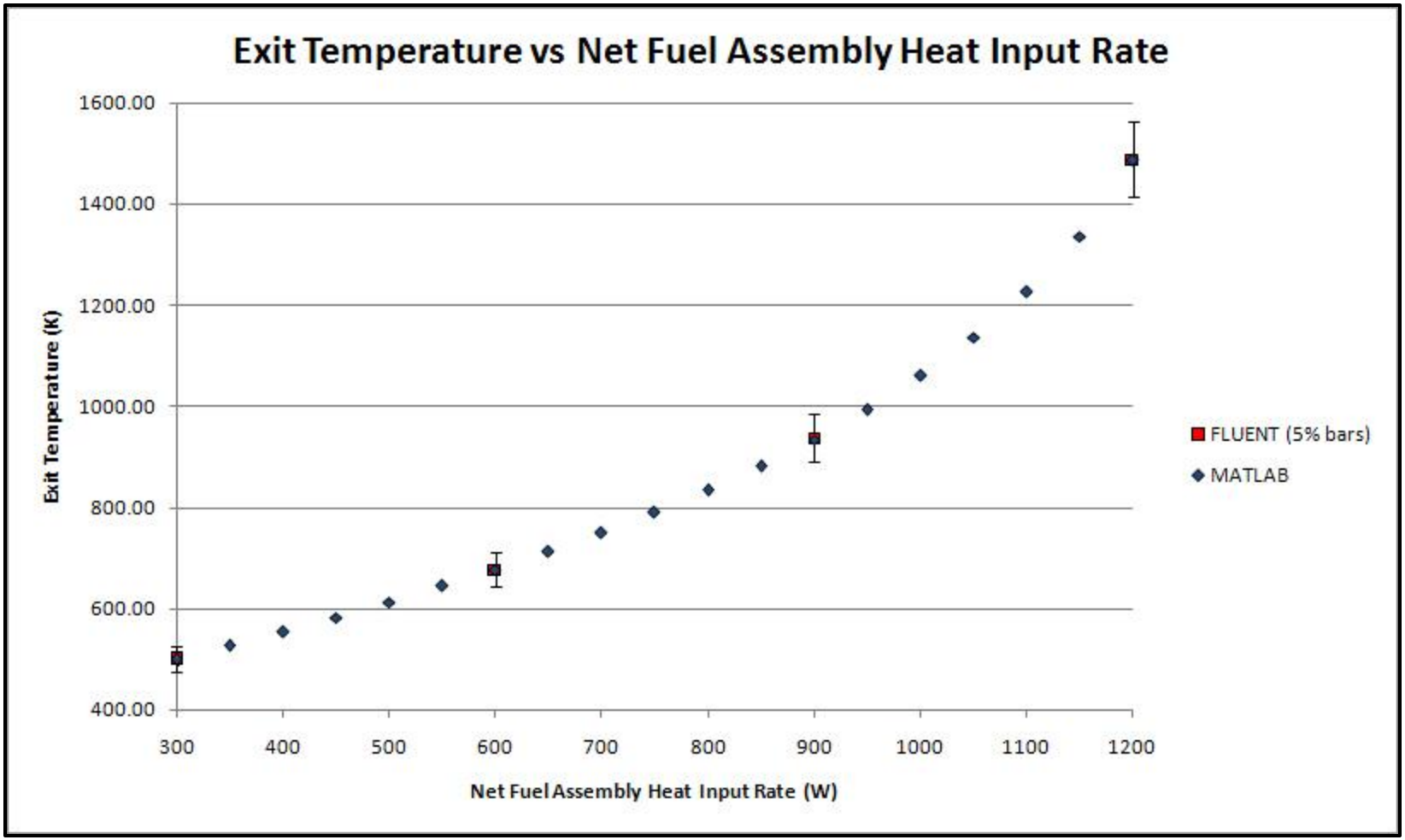


Figure 6.7: Comparison of MATLAB program base trials for exit temperature versus net heat input rate with FLUENT CFD benchmark values determined from heated, buoyancy-driven flow simulations.

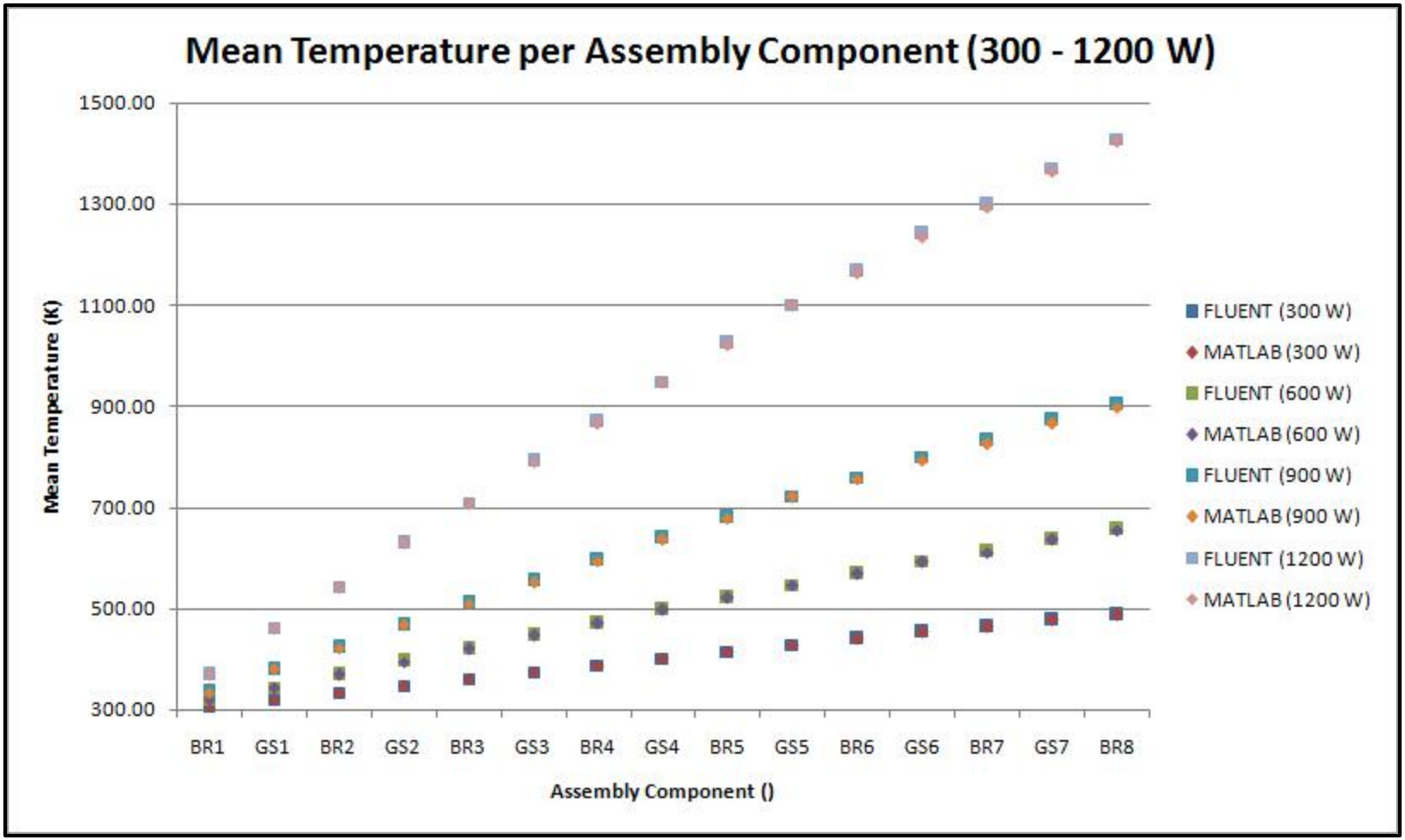


Figure 6.8: Comparison of MATLAB program base trials for component (flow path segment) mean temperatures at specific net heat input rates with FLUENT CFD benchmark values determined from heated flow simulations.

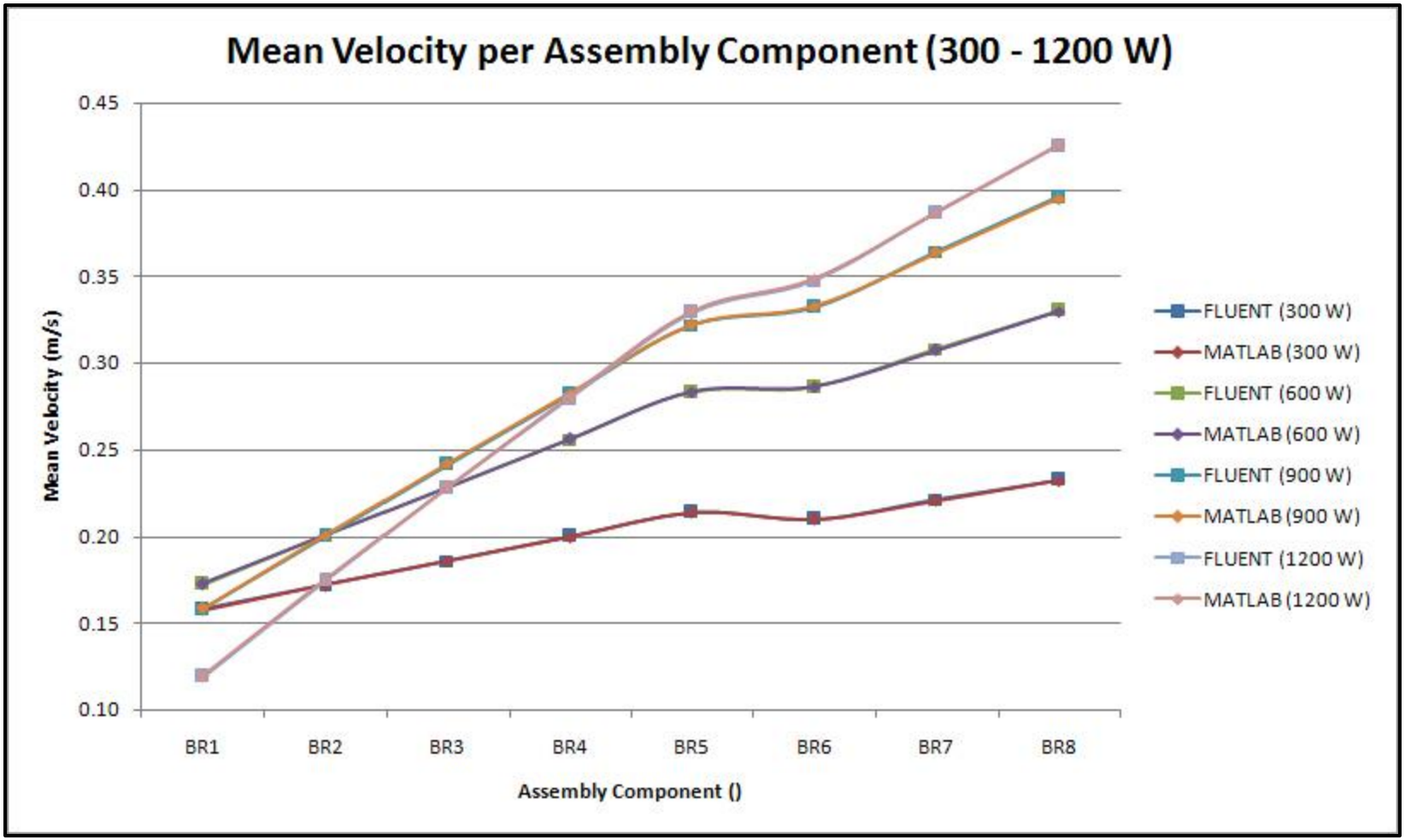


Figure 6.9: Comparison of MATLAB program base trials for bundle run component (flow path segment) mean velocities at specific net heat input rates with FLUENT CFD benchmark values determined from heated flow simulations.

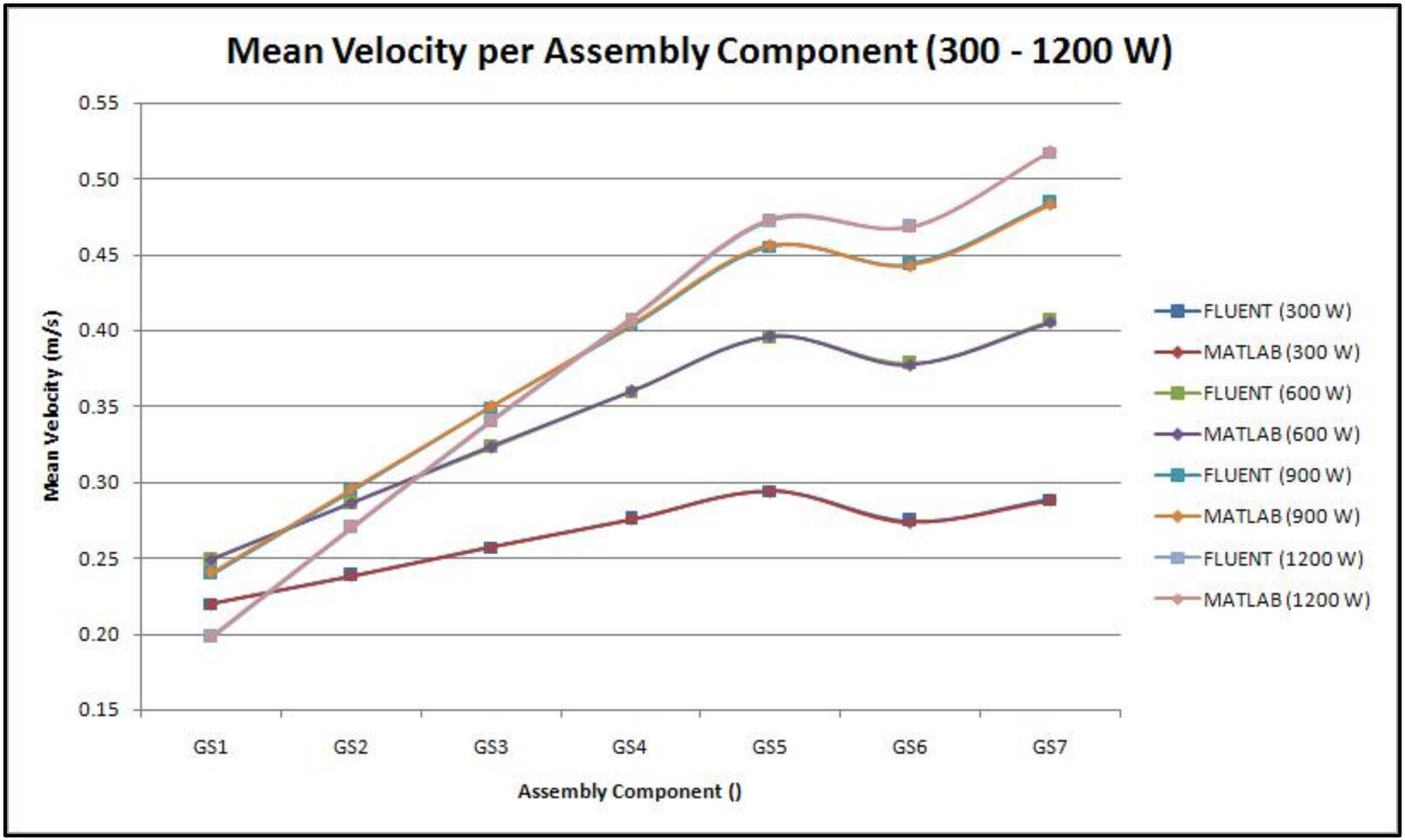


Figure 6.10: Comparison of MATLAB program base trials for spacer component (flow path segment) mean velocities at specific net heat input rates with FLUENT CFD benchmark values determined from heated flow simulations.

Table 6.3: Comparison of numerical solution data from MATLAB program base trials with FLUENT CFD benchmark values from heated, buoyancy-driven flow simulations (1 of 2).

COMPARISON OF NUMERICAL SOLUTION DATA FROM MATLAB PROGRAM BASE TRIALS WITH FLUENT CFD BENCHMARK VALUES FROM HEATED FLOW SIMULATIONS

[Mass Flow Rate]

Trial	FLUENT (kg/s):	MATLAB (kg/s):	Difference (%):
300 Watts	0.00144046	0.00143845	0.139231
600 Watts	0.00151808	0.00151951	0.094013
900 Watts	0.00132222	0.00132961	0.558668
1200 Watts	0.00089970	0.00090466	0.550722

[Total Pressure Loss]

Trial	FLUENT (Pa):	MATLAB (Pa):	Difference (%):
300 Watts	9.497589	9.528770	0.328302
600 Watts	14.176700	14.197900	0.149538
900 Watts	18.562453	18.533400	0.156516
1200 Watts	23.792835	23.784600	0.034612

[Fuel Assembly Exit Temperature]

Trial	FLUENT (K):	MATLAB (K):	Difference (%):
300 Watts	499.6686	499.9730	0.060924
600 Watts	677.9746	677.6280	0.051115
900 Watts	937.8379	933.9360	0.416052
1200 Watts	1488.4552	1486.5800	0.125983

[Bundle Run Component (Flow Path Segment) Mean Temperatures]

Source / Trial	BR1 (K):	BR2 (K):	BR3 (K):	BR4 (K):	BR5 (K):	BR6 (K):	BR7 (K):	BR8 (K):
FLUENT - 300 Watts	307.1983	334.3297	361.4125	388.4297	415.3636	440.8883	464.7486	488.4943
MATLAB - 300 Watts	307.1800	334.3290	361.4740	388.5530	415.5490	441.0540	464.0140	487.8720
Difference (%)	**0.005951**	**0.000218**	**0.017008**	**0.031748**	**0.044643**	**0.037588**	**0.158068**	**0.127400**
FLUENT - 1200 Watts	370.6160	543.2896	709.5402	869.5100	1024.2838	1167.7781	1299.4472	1428.3858
MATLAB - 1200 Watts	370.2050	542.0840	707.7380	867.1900	1021.5600	1164.5100	1293.5300	1423.7000
Difference (%)	**0.110894**	**0.221909**	**0.253988**	**0.266817**	**0.265922**	**0.279856**	**0.455363**	**0.328049**

Table 6.4: Comparison of numerical solution data from MATLAB program base trials with FLUENT CFD benchmark values from heated, buoyancy-driven flow simulations (2 of 2).

[Spacer Component (Flow Path Segment) Mean Temperatures]							
Source / Trial	GS1 (K):	GS2 (K):	GS3 (K):	GS4 (K):	GS5 (K):	GS6 (K):	GS7 (K):
FLUENT - 300 Watts	320.8070	347.9110	374.9593	401.9335	428.8163	453.8829	477.6737
MATLAB - 300 Watts	320.7750	347.9420	375.0520	402.0870	429.0370	453.1280	476.9910
Difference (%)	**0.009981**	**0.008916**	**0.024717**	**0.038203**	**0.051474**	**0.166329**	**0.142918**
FLUENT - 1200 Watts	462.1247	630.3054	792.5037	949.1902	1101.4811	1240.9632	1371.0587
MATLAB - 1200 Watts	461.2150	628.7800	790.4140	946.6330	1098.6300	1233.9100	1364.6600
Difference (%)	**0.196860**	**0.242002**	**0.263677**	**0.269404**	**0.258842**	**0.568365**	**0.466698**

[Bundle Run Component (Flow Path Segment) Mean Velocities]								
Source / Trial	BR1 (m/s):	BR2 (m/s):	BR3 (m/s):	BR4 (m/s):	BR5 (m/s):	BR6 (m/s):	BR7 (m/s):	BR8 (m/s):
FLUENT - 300 Watts	0.158080	0.172041	0.185978	0.199880	0.213740	0.210059	0.221428	0.232741
MATLAB - 300 Watts	0.157860	0.171834	0.185805	0.199741	0.213635	0.209954	0.220894	0.232261
Difference (%)	**0.138973**	**0.120396**	**0.092773**	**0.069623**	**0.049099**	**0.050154**	**0.240941**	**0.206263**
FLUENT - 1200 Watts	0.119119	0.174617	0.228052	0.279467	0.329213	0.347514	0.386697	0.425067
MATLAB - 1200 Watts	0.119634	0.175165	0.228685	0.280202	0.330079	0.348376	0.386970	0.425912
Difference (%)	**0.432504**	**0.313586**	**0.277737**	**0.262948**	**0.263165**	**0.247960**	**0.070553**	**0.198678**

[Spacer Component (Flow Path Segment) Mean Velocities]							
Source / Trial	GS1 (m/s):	GS2 (m/s):	GS3 (m/s):	GS4 (m/s):	GS5 (m/s):	GS6 (m/s):	GS7 (m/s):
FLUENT - 300 Watts	0.220072	0.238665	0.257220	0.275724	0.294165	0.274234	0.288609
MATLAB - 300 Watts	0.219772	0.238412	0.257012	0.275561	0.294051	0.273545	0.287964
Difference (%)	**0.136136**	**0.105898**	**0.080761**	**0.059050**	**0.038843**	**0.251345**	**0.223328**
FLUENT - 1200 Watts	0.198006	0.270066	0.339563	0.406698	0.471950	0.468312	0.517408
MATLAB - 1200 Watts	0.198682	0.270852	0.340469	0.407754	0.473219	0.468111	0.517715
Difference (%)	**0.341429**	**0.291016**	**0.266830**	**0.259603**	**0.268866**	**0.043007**	**0.059410**

The graphical results in Figures 6.5 through 6.10 as well as numerical values in Tables 6.3 and 6.4 impart a considerable wealth of solution data essential in completing a thorough comparative analysis with the benchmark CFD simulations presented in Section 5.3.3, which is necessary for establishing the overall validity of the MATLAB computational program. Although a number of inferences are quite evident as a result of the above comparisons, there are a few key conclusions of primary interest. These deductions are briefly summarized in the bulleted list below.

- Foremost, the MATLAB program results exhibit a tremendous degree of accuracy in comparison to the FLUENT CFD solutions for every parameter of significance in judging the thermal hydraulic response of the fuel assembly system. The difference in predicted values for each parameter compared is well within a 1% margin of error.
- Despite the use of a single set of initial mean temperature estimates for all computational trials over a broad range of net heat input rates, the accuracy of determined parameter values illustrates negligible effect on the solution outcome. This desirable characteristic of the inherent code is just one of several sensitivity parameters studied in assessing the application robustness of the MATLAB program as continued in Section 6.4.2 below.
- Although increasing net heat input rates inevitably translate into greater temperature differences between internal and external air conditions (providing stronger buoyancy forces to drive the flow), a limit develops near the heat input rate of approximately 500 watts where the mass flow rate reaches a maximum and subsequently declines with further increase in the net heat input rate value. This is important because it signifies a point upon which increasing effects from higher

dynamic viscosities on flow resistances become substantial with regard to the overall temperature response inside the fuel assembly and net changes in the specific heat capacity and dynamic viscosity properties of air are critically influential in determining the maximum temperature that develops near the outlet.

- Last, but certainly not least, the average wall-clock time required to setup and complete a calculation using the MATLAB program is less than 45 seconds from start to finish on a standard 2 GHz laptop computer. This is an extraordinary benefit in computational efficiency over the FLUENT CFD simulations described in Section 5.3.3, which on average required approximately two weeks of research time for model setup, submission/monitoring of repeated job sets for convergence, and postprocessing of final solution values.

6.4.2 Parameter Sensitivity Trials

A second phase of research trials undertaken in order to establish a reasonable level of confidence in the validity of this computational code development, focused on several input parameter sensitivity studies intended to evaluate the range of suitable application for general employment of the expressed program capabilities. The method of approach used in selecting the specific sequence of parameter sensitivities examined in these studies is purposely aimed at reflecting highly probable situations of program application for analyzing the base setup detailed in Tables 6.1 and 6.2, but with a progressively reduced extent of reference experimental/supported theoretical data available to the researcher for performing the desired computation. In other words, in addition to the initial circumstance pertaining to a lack of sufficient information for entering a starting set of mean temperature estimates as incorporated in the

aforementioned base comparisons, each succeeding sensitivity trial below necessitates the assumption of at least one further parameter input on behalf of the researcher. The values selected for these parameters are determined according to reasonable, best-known approximations or sources of numerically tabulated data from widely available reference publications.

Unless otherwise noted, the default characteristic input values, initial estimates, and solution controls are assumed as listed for the base trials in Tables 6.1 and 6.2 above. Due to the vast number of solution values available as output from the MATLAB program, only the most notable parameter results are presented as convincingly appropriate for the comparative analysis. Brief descriptions of the sensitivity studies performed and correlating final parameter solutions obtained are provided in order of decreasing reference experimental/supported theoretical data availability below.

- **Sensitivity Study #1** - assessment of solution accuracy for circumstances characterized by an uncertainty in the initial mean temperature estimate for each flow path segment. For example, the experimental results for a representative buoyancy-driven flow are available as a reference to all necessary input parameter values, but the net heat input rate under consideration is largely different from any point in the experimental data set. Thus, the sensitivity of initial mean temperature estimates on the solution accuracy is particularly important in such instances. This requires an examination of various differences between initial estimate inputs and final steady-state values for the mean temperatures of all flow path segments on the predicted solution outcome. An approximation for the average variance in potential input values reasonably encountered across the range of net heat input rates considered is delegated for each flow path segment as

the average value over all mean temperature data points (for that segment) from the heated CFD simulations. As previously discussed, the results of this sensitivity study are inherently illustrated as part of the base comparison trials in Figures 6.5 through 6.10 as well as Tables 6.3 and 6.4 above; correlating parameter values from these base comparison trials are included with the findings of the following sensitivity studies for additional reference.

- **Sensitivity Study #2** - assessment of solution accuracy for circumstances characterized by the previous sensitivity study with additional uncertainty in the values for stratification parameters of flow path segments. For example, the experimental pressure loss results from a representative buoyancy-driven flow are available as a reference to necessary input values for the hydraulic loss coefficients, but the experimental data set does not include detailed measurements of the developed temperature distributions along the fuel assembly length. Hence, it is of utmost interest to evaluate the sensitivity of the solution accuracy to a best estimate of the stratification parameter values as requisite in cases lacking any available reference data for the temperature distributions.

  A practical approach for approximating the stratification parameter values in this analysis stems from the sensible assumption of a uniform heat flux distribution across the fuel assembly. As previously discussed, published research efforts involving buoyancy-induced airflows through tall heated ducts of uniform surface heat flux reveal a linear temperature distribution development (*i.e.* $S = 0.5$) along the duct length (Brinkworth *et al.*, 2000; Brinkworth, 2000). Therefore, a reasonable estimate of the stratification parameter value for each flow path

segment in this sensitivity study assessment is ($S = 0.5$) and the resulting accuracy of the solution values are compared in Figures 6.11 through 6.13 below.

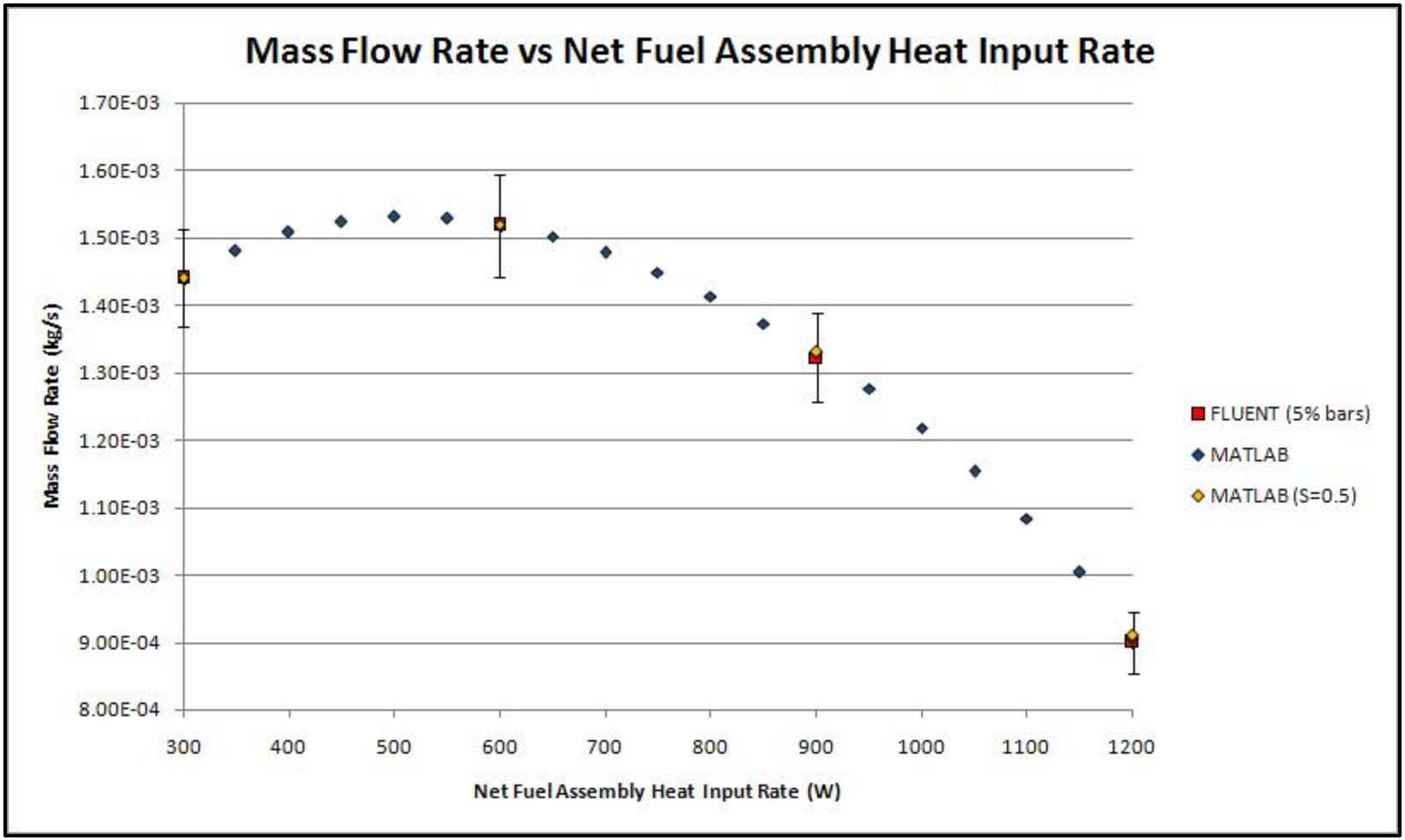


Figure 6.11: Comparison of MATLAB program sensitivity study #2 results for mass flow rate versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

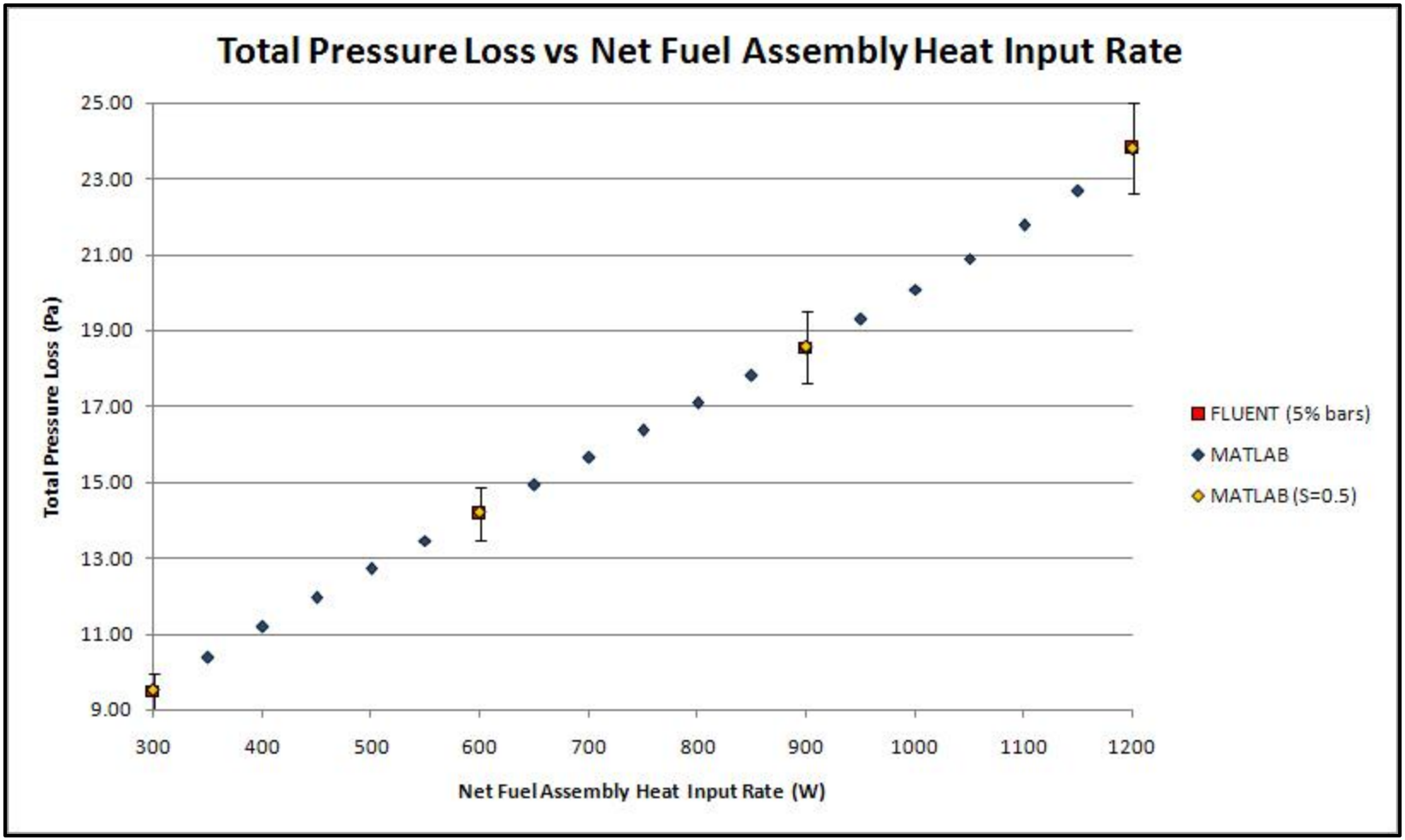


Figure 6.12: Comparison of MATLAB program sensitivity study #2 results for total pressure loss versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

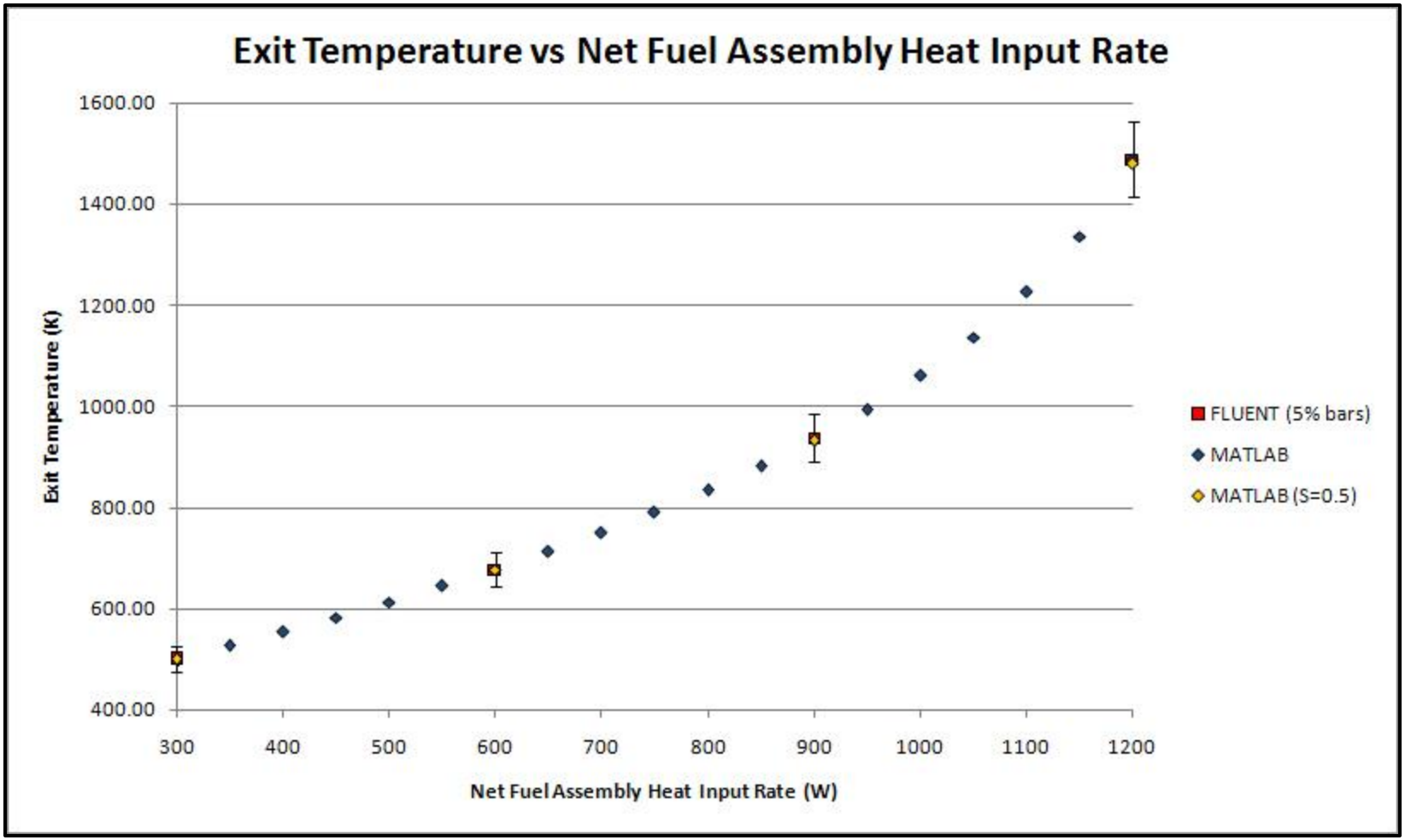


Figure 6.13: Comparison of MATLAB program sensitivity study #2 results for exit temperature versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

- **Sensitivity Study #3** - assessment of solution accuracy for circumstances characterized by the previous sensitivity study with additional uncertainty in the hydraulic loss coefficient values of flow path segments for buoyancy-driven flows. For example, experimental pressure loss results for airflows through the fuel assembly are available as a reference to necessary input values for the hydraulic loss coefficients, but the experimental data set is based on unheated, forced flow conditions rather than buoyancy-induced forces within an internally heated airflow. The sensitivity of hydraulic loss coefficients derived from unheated, forced flow experimental data on the accuracy of solutions obtained for heated, buoyancy-driven flow computations using the MATLAB program is of pertinent relevance to application of the code in such circumstances.

  As a result, the unheated, forced flow loss coefficients from the FLUENT CFD simulations in Section 5.3.2 are implemented into the program code for the current buoyancy-driven flow analyses and the parameter solution values are compared in order to evaluate the influence on resulting accuracy. The hydraulic loss coefficients for each of the flow path segments are provided in Table 6.5 followed by a comparison of significant parameter solutions in Figures 6.14 through 6.16 below.

Table 6.5: Hydraulic loss coefficients for flow path segments used in sensitivity study #3; based on FLUENT CFD simulations from Section 5.3.2 for unheated, pressure-driven flows using the equivalent assembly model.

FLUENT Equivalent Fuel Assembly Model: TableCurve 2D Curve-Fit Data For Hydraulic Loss Coefficients (Unheated)					
[BR1-Inlet-Segment]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.941034	0.030252	1.0000	96.7569	0.061650
[BR-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.969000	0.267182	1.0000	97.6770	0.544484
[GS-Fully-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.169390	0.075111	1.0000	66.1042	0.153067
[BR-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	2.200103	0.253063	0.9993	101.0240	0.515710
[GS-Partially-Populated-Segments]	a1:	a2:	R^2:	SLAM:	k:
200 - 600 (slpm)	0.934320	0.151016	1.0000	43.4322	0.307751

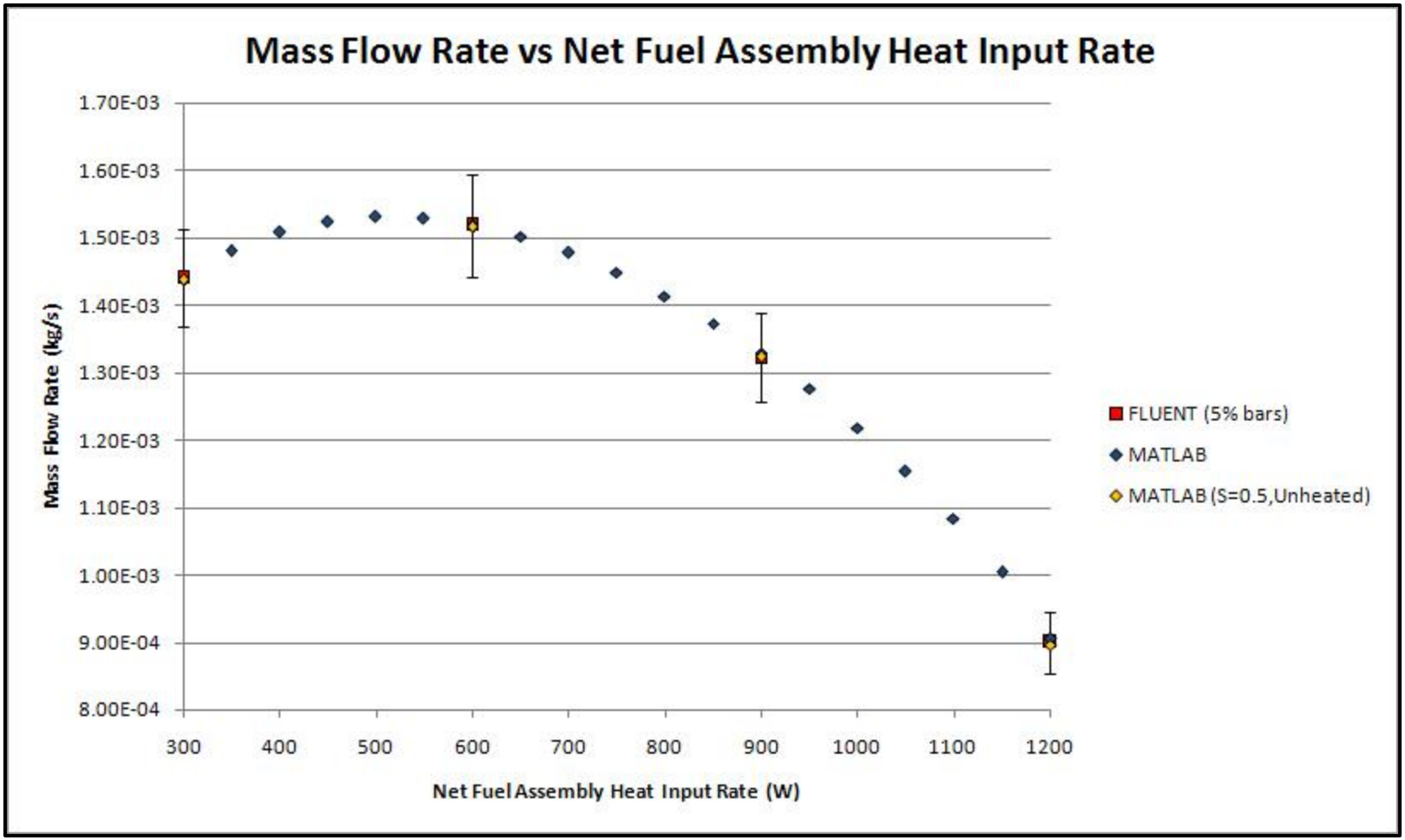


Figure 6.14: Comparison of MATLAB program sensitivity study #3 results for mass flow rate versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

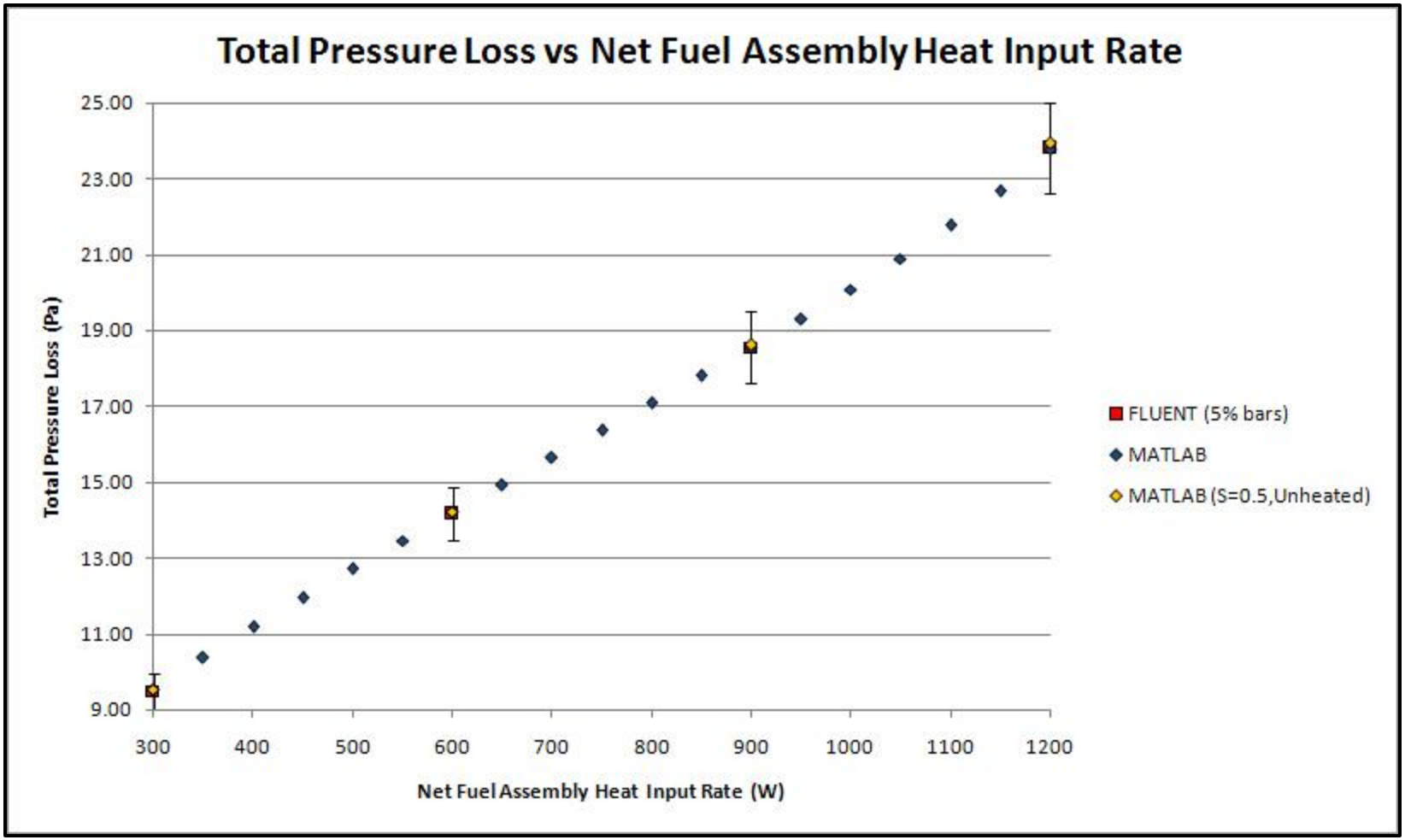


Figure 6.15: Comparison of MATLAB program sensitivity study #3 results for total pressure loss versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

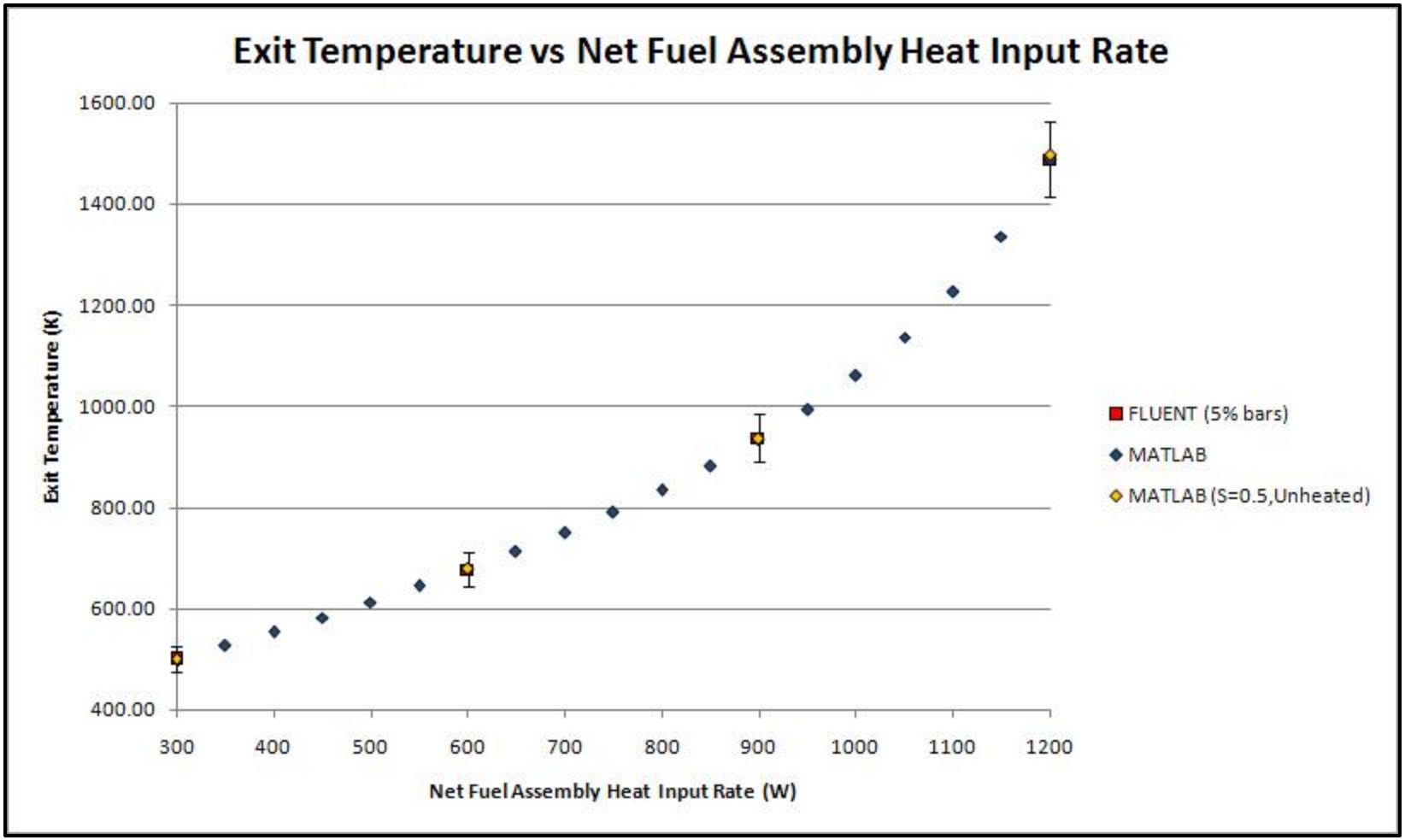


Figure 6.16: Comparison of MATLAB program sensitivity study #3 results for exit temperature versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

- **Sensitivity Study #4** - assessment of solution accuracy for circumstances characterized by the previous sensitivity study with additional universal uncertainty in the hydraulic loss coefficient values of flow path segments. For example, this corresponds to the application of the program code in a situation where any reference experimental/supported theoretical data facilitating the identification of reasonably validated inputs for required parameter values is nonexistent. Thus, the researcher is responsible for providing best estimate values for all principal parameter inputs. Sensitivity of the solution accuracy to a scenario in which reasonable, but unproven approximations are incorporated into the computational analysis for every key input parameter is of primary interest in this evaluation. The input estimates for the initial mean temperatures and stratification terms remain the same as specified in the first two sensitivity studies; however, rational approximations for the viscous (S_{LAM}) and form (k) loss coefficients for each flow path segment are necessary.

  Pertaining to the form loss coefficients, average approximations over adjacent flow path segments for the contraction/expansion loss coefficient values associated with interconnecting segments are utilized to derive the form loss coefficient final estimates. These contraction/expansion loss coefficient values are represented in graphical format based on the interconnecting area ratio of adjacent segment flow areas and widely available in most general fluid mechanics texts (Fox and McDonald, 1985). In addition, the viscous loss coefficients for flow path segments are simply approximated from the nearest representations of the cross-sectional flow area shape (remembering, the equivalent fuel assembly CFD model in Section 5.3 is being compared) to several general cross-section shapes tabulated with accompanying viscous loss coefficient values as available

in general convection heat transfer texts (Incropera and DeWitt, 2002). However, the cross-section and viscous loss coefficient for the partially-populated spacers are an exception that requires further consideration.

Unfortunately, the cross-sectional area of flow associated with the partially-populated spacer flow segments in the equivalent assembly model is comprised of more than one wall-bounded flow pathway per cross-section segment of repeated symmetry, as illustrated in Section 5.3 above. In other words, the cross-section of the partially-populated spacer component is not singly-connected as it contains two flow branches per segment of symmetry. Although numerous publications exist on proposed methods for estimating viscous loss coefficients in ducted flows of arbitrary cross-sections, these methods are only applicable to singly-connected cross-sections. To the best of the author's knowledge, no standard published approach exists for estimating the equivalent viscous loss coefficient of an arbitrary multipath cross-sectional area of flow.

A working hypothesis is therefore necessary at this point, without immediate justification (until a latter section of this chapter), in order to proceed with the intended focus of the current sensitivity study. Consequently, it is postulated that the equivalent viscous loss coefficient value for the multipath cross-section of the partially-populated spacer components is represented by an expression of the form akin to the equivalent resistance of a simple electrical circuit consisting of resistors connected in parallel (one parallel resistor for each distinct branch/path of flow traversing the cross-section). Furthermore, an extension of the same analysis used in the evaluation of specific parameter values in a simple electrical circuit loop is also analogously employed in the calculation of resulting viscous loss coefficients for multipath cross-sectional areas of flow. This estimation

method is addressed in substantial detail and subjected to a number of accuracy validations as well as sensitivity studies in Section 6.4.3. However, with the final reasonable estimate of the viscous loss coefficient for the partially-populated spacer segments, the viscous and form loss coefficients represented in Table 6.6 are implemented and resulting significant parameter solutions compared in Figures 6.17 through 6.19 below.

Table 6.6: Hydraulic loss coefficients for flow path segments used in sensitivity study #4; values represent best estimates/approximations from reference data available in general textbook/research publications (excluding the partially-populated spacer segment, see Section 6.4.3).

Sensitivity Study #4 Hydraulic Loss Coefficients - Best Estimates/Approximations Based On General Textbook/Research Publications

{See Section 6.4.3 For Partially-Populated Spacer Estimate Using Equivalent Resistance Method}

[Viscous Loss Coefficients - S_{LAM}]

[BR1-Inlet]	X-Section Shape:	Side Length Ratio:	Reference:	SLAM:
Details/Notes/Values	Parallel Plates	~ 273.3 → Infinite	Incropera & DeWitt (2002)	~ 96.0
[BR-Fully-Populated]	X-Section Shape:	Side Length Ratio:	Reference:	SLAM:
Details/Notes/Values	Parallel Plates	~ 273.3 → Infinite	Incropera & DeWitt (2002)	~ 96.0
[GS-Fully-Populated]	X-Section Shape:	Side Length Ratio:	Reference:	SLAM:
Details/Notes/Values	Rectangular Tube	~ 2.614	Incropera & DeWitt (2002)	~ 66.0
[BR-Partially-Populated]	X-Section Shape:	Side Length Ratio:	Reference:	SLAM:
Details/Notes/Values	Parallel Plates	~ 211.5 → Infinite	Incropera & DeWitt (2002)	~ 96.0
[GS-Partially-Populated]	X-Section Shape:	Side Length Ratio:	Reference:	SLAM:
Details/Notes/Values	Parallel Plates (Out)	~ 458.8 → Infinite	Incropera & DeWitt (2002)	~ 96.0
	Rectangular Tube (In)	~ 1.457	Incropera & DeWitt (2002)	~ 59.0
(See Section 6.4.3)			(SLAM Equivalent) →	~ 44.0

[Form Loss Coefficients (Same Across Segment Types) - k]

The sum of expansion/contraction losses due to a spacer segment is equally attributed between adjacent fuel bundle and spacer segments. An average x-section area ratio (GS/BR) over the range of values from all interconnecting flow path segments across the fuel assembly is used in a simple estimation of these losses.

[All Segments]

X-Section Area Ratio Range (GS/BR):	Average Area Ratio:
~ 0.69 - 0.79	0.74

Expansion/Contraction Loss Sum (k):	Reference:
~ 0.25	Fox & McDonald (1985)

k (per bundle):	k (per spacer):
~ 0.125	~ 0.125

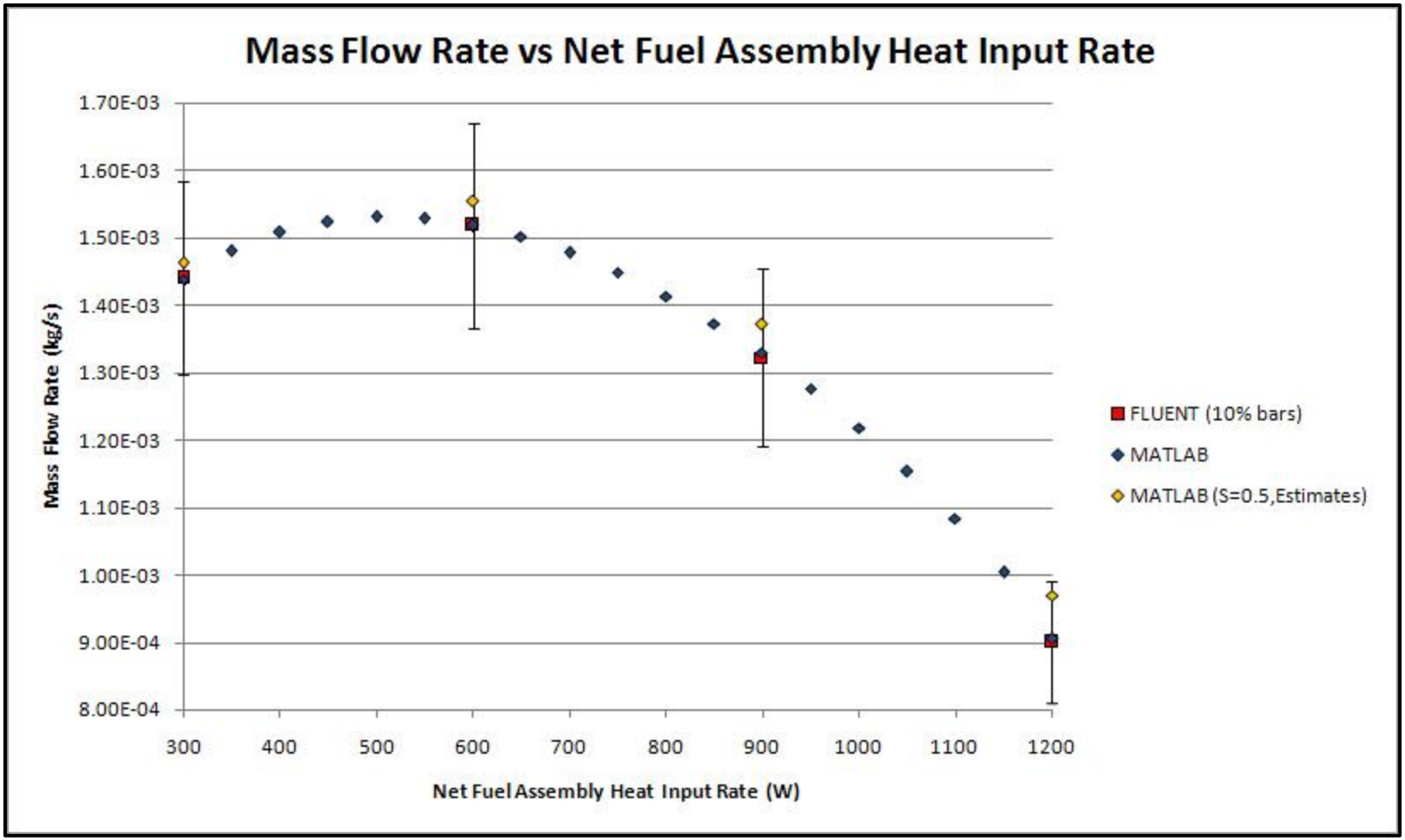


Figure 6.17: Comparison of MATLAB program sensitivity study #4 results for mass flow rate versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

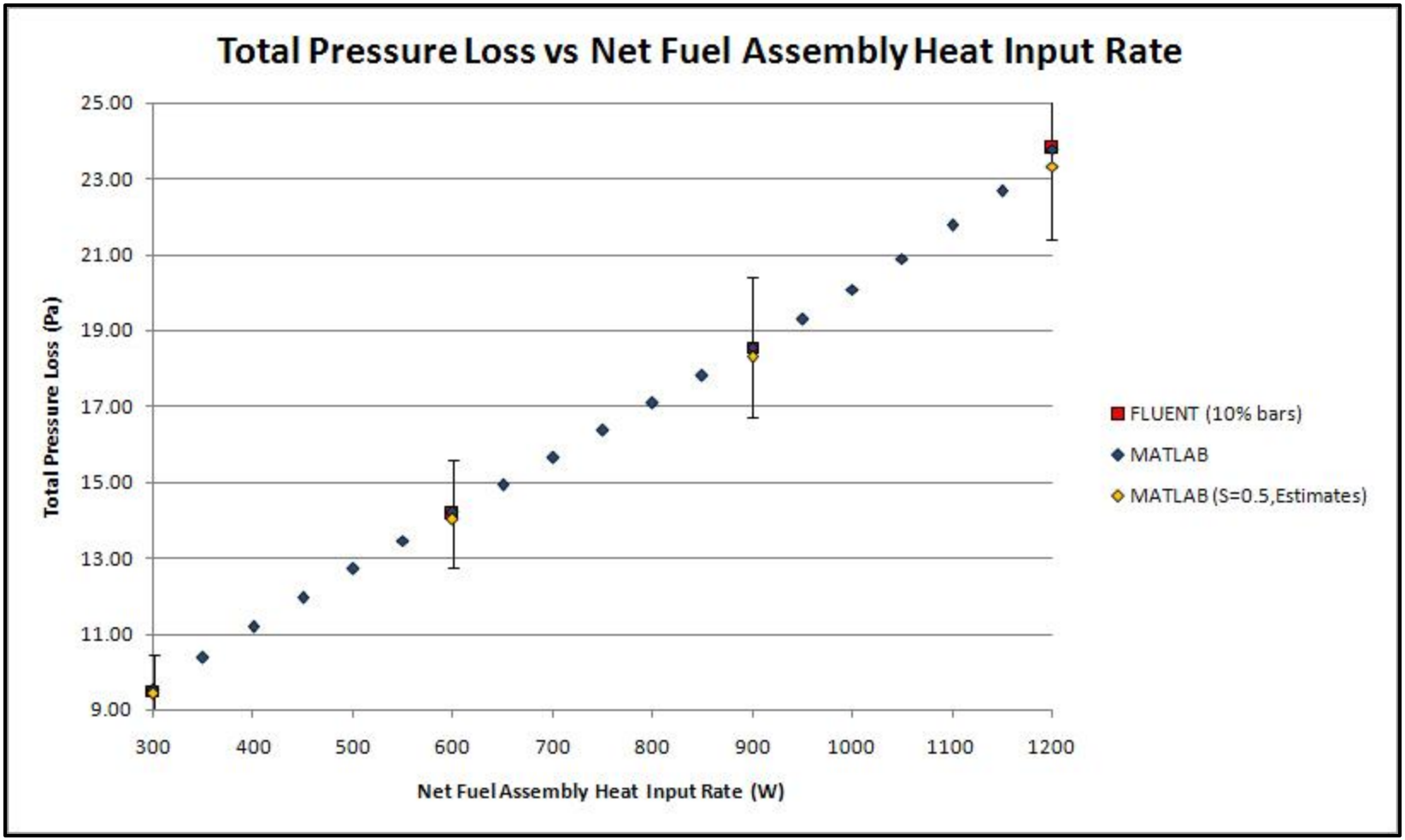


Figure 6.18: Comparison of MATLAB program sensitivity study #4 results for total pressure loss versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

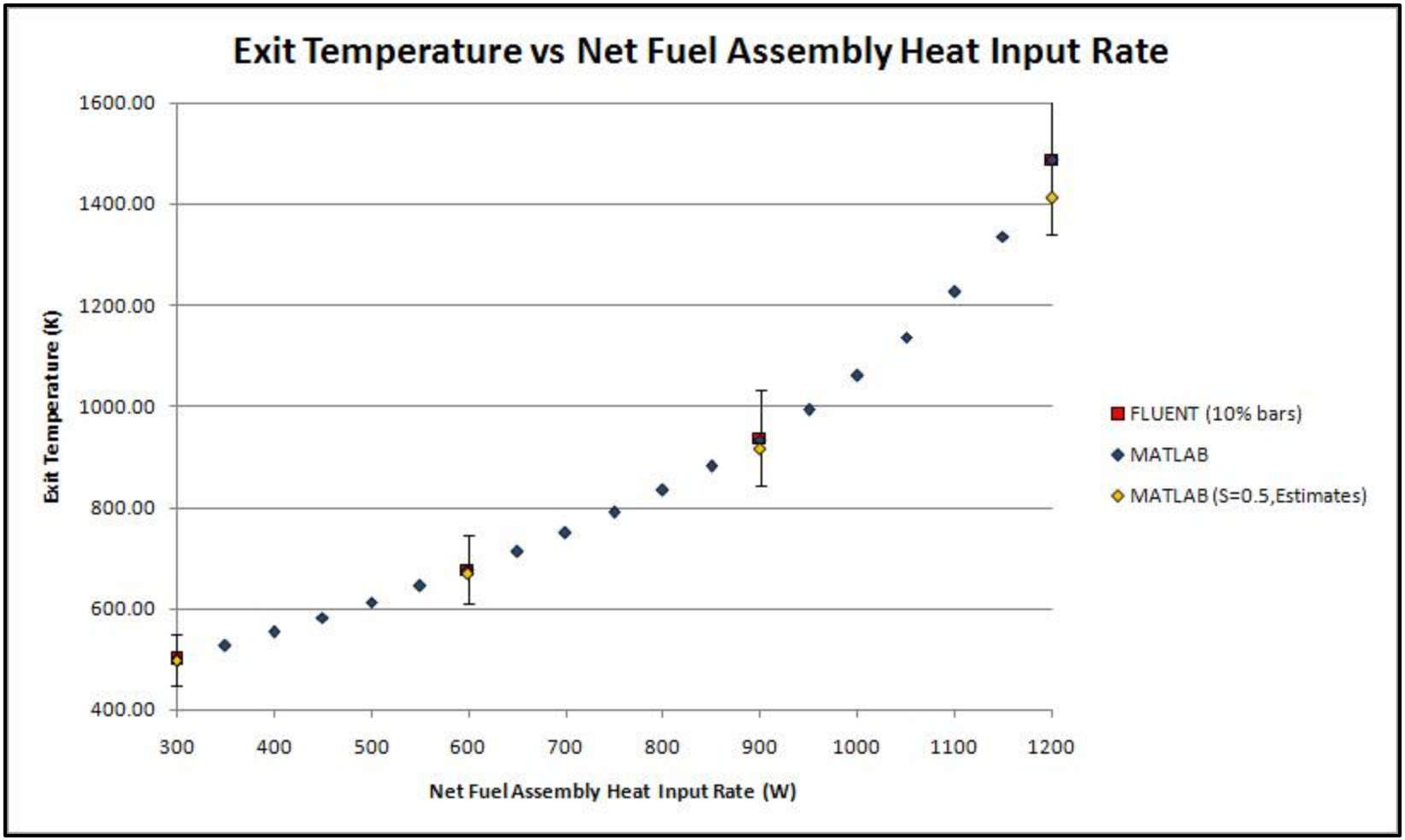


Figure 6.19: Comparison of MATLAB program sensitivity study #4 results for exit temperature versus net heat input rate with MATLAB program base trials and FLUENT CFD benchmark values.

The comparison of results from the sensitivity trials above clearly demonstrates a broad range of application scenarios in which the MATLAB computational analysis program provides accurate solutions despite varying extents of reference data availability for the required input parameters. As long as reasonable estimates are provided for the unknown parameter values, sensitivity studies #1 through #3 illustrate a general presumption that solutions obtained for all key variable outputs are likely within an approximate 2% margin of error. This is particularly important because it suggests that less expensive, unheated experiments involving pressure-driven flows are sufficient for obtaining the hydraulic loss coefficients since the results acquired using both unheated- and heated-based experimental loss coefficients are comparable. Furthermore, even in situations lacking any experimental reference data for selecting the input parameter values, the comparison of results from sensitivity study #4 shows that solutions within a 10% margin of error are still achievable if appropriate values based on reference publications are implemented. The equivalent resistance method for estimating the viscous loss coefficient utilized in sensitivity study #4 for the partially-populated spacer segments is subsequently addressed in the following discussion.

6.4.3 Approximation Method for Multipath Flow Segments

Approximating the viscous loss coefficient for a component cross-sectional area comprised of multiple flow branches, as attributed to the partially-populated spacer segments above, presents a uniquely difficult situation for determining appropriate input values for the MATLAB program when specific experimental reference data is unavailable. Although numerous publications on proposed methods for determining viscous (friction) loss coefficients in ducted flows of arbitrary cross-sections exist, these

methods are only applicable to singly-connected cross-section paths. Therefore, due to the absence of a supported, standard published approach (to the best of the author's knowledge), a computational analysis such as sensitivity study #4 requires a reasonable method for estimating the friction loss factor for flow segments with a multiple path cross-section. Using the calculation in sensitivity study #4 for partially-populated spacer segments from the equivalent assembly model as an example, a simple process for approximating this loss coefficient is proposed for consideration below.

Several assumptions are necessary from a practical standpoint for logical justification of the methodology employed; however, it is certainly possible that values obtained under these assumptions are applicable in many other situations as well. Nonetheless, these assumptions are briefly outlined as follows:

1. Flow is assumed laminar, incompressible, and steady with constant fluid properties. Expected behavior is characterized by a small fluid velocity and low Reynolds number in which the friction factor derivation assumes an unheated flow field.
2. The pressure losses in flow branches are dominated by viscous effects, which are linear by definition and solely responsible for the resulting flow rate in each branch directly. This is a reasonable assumption noting the dominance of the linear loss term in flows with a low fluid velocity and Reynolds number.
3. As an extension to the above assumption, minor losses are assumed to have negligible effect on the flow rate division in each branch although these losses are still included as part of the total loss attributed to the entire flow path segment in the MATLAB program calculations.

The fundamental basis for the proposed method follows from a presumption that flow through the multipath cross-section of a partially-populated spacer segment is effectively represented as a simple electrical circuit consisting of two resistors (one for each flow path branch) connected in parallel, with a potential difference maintained across the spacer element. A diagram of the spacer segment cross-sectional area of flow from the equivalent assembly model is illustrated in Figure 6.20 below along with the representative electrical circuit drawing.

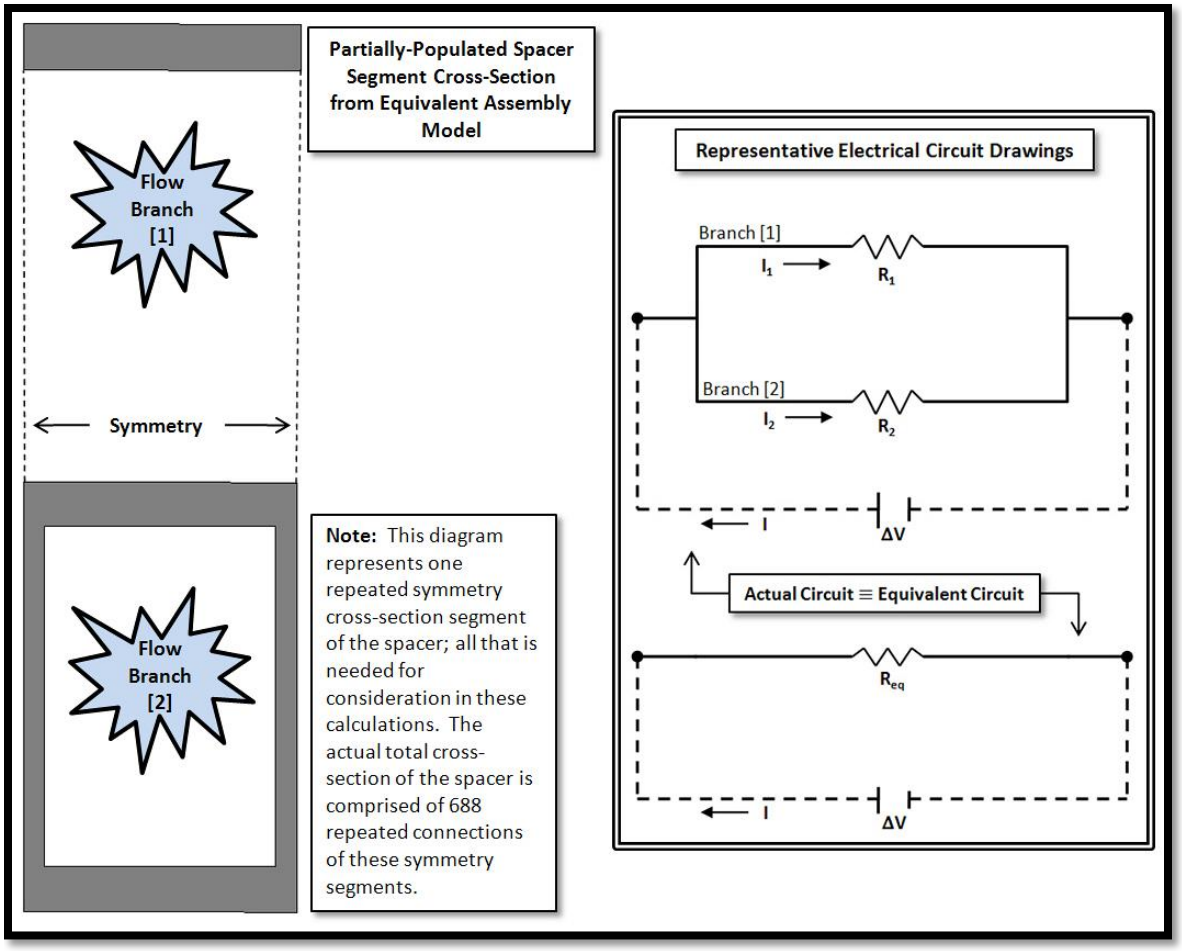


Figure 6.20: Diagram of spacer cross-sectional area of flow from equivalent assembly model with representative electrical circuit drawings.

Using widely known relationships for parallel resistors in a simple circuit in conjunction with Kirchhoff's rules, the characteristic equations for the representative electrical circuit drawings in Figure 6.20 are stated as follows:

$$I = I_1 + I_2 \tag{6.50}$$

$$\Delta V = I_1 R_1 = I_2 R_2 \tag{6.51}$$

$$\Delta V = I R_{eq} \tag{6.52}$$

where, (I) and (R) are the current and resistance, respectively as designated in the circuit diagram, and (ΔV) is the potential across the entire circuit. Thus, from the relationships given by Equations (6.50) through (6.52) above, the final governing circuit equation is written as

$$\frac{\Delta V}{R_1} + \frac{\Delta V}{R_2} = \frac{\Delta V}{R_{eq}} \tag{6.53}$$

Furthermore, considering the aforementioned assumptions and realizing Kirchhoff's first rule is simply a conservation statement of electric charge (mechanically analogous to conservation of mass) and that Kirchhoff's second rule is a conservation statement of energy/electric potential across all elements around any closed circuit loop (mechanically analogous to driving potential and losses across a component segment), reveals the following equivalences:

$$I(current) \equiv Q(flow\ rate) \tag{6.54}$$

$$\Delta V(potential) \equiv \Delta P(pressure) \tag{6.55}$$

This leaves the resistance (R) as the only variable without a direct correlation to a mechanically equivalent parameter definition. However, from the two mechanically analogous parameters defined above [Equations (6.54) and (6.55)] and stressing again the previously listed assumptions (especially the negligible effects of minor/form losses in the system on resulting flow rate through either flow branch), an expression for the mechanical equivalent of electrical resistance is derived as follows:

$$\Delta V = IR(electrical) \equiv \Delta P = QR(mechanical) \tag{6.56}$$

where, from the definition of major head loss, the differential pressure (ΔP) is given by

$$\Delta P = f\left(\frac{L}{D_h}\right)\left(\frac{\rho V^2}{2}\right) = \frac{S_{LAM} L\mu}{2D_h^2}(V) \tag{6.57}$$

An expression for the flow rate is also simply written as

$$Q = VA_c \tag{6.58}$$

Substituting Equation (6.58) into Equation (6.57) for the velocity, yields

$$\Delta P = \frac{S_{LAM} L\mu}{2D_h^2 A_c}(Q) \tag{6.59}$$

and, by incorporating this expression for (ΔP), Equation (6.56) becomes

$$QR(mechanical) = \frac{S_{LAM} L\mu}{2D_h^2 A_c}(Q) \tag{6.60}$$

Therefore, the analogous expression for the electrical resistance as a mechanical parameter is given by

$$R(mechanical) = \frac{S_{LAM} L\mu}{2D_h^2 A_c} \tag{6.61}$$

In summary, the equivalent mechanical parameters for resistances in the electrical circuit drawings from Figure 6.20 are easily expressed using the form of Equation (6.61) as follows:

$$R_1 = \frac{S_{LAM1} L_1 \mu_1}{2{D_{h1}}^2 A_{c1}} \tag{6.62}$$

$$R_2 = \frac{S_{LAM2} L_2 \mu_2}{2{D_{h2}}^2 A_{c2}} \tag{6.63}$$

$$R_{eq} = \frac{S_{LAMeq} L\mu}{2{D_{heq}}^2 (A_{c1} + A_{c2})} \tag{6.64}$$

where, the variables in Equations (6.62) through (6.64) are characterized/defined as

$$L = L_1 = L_2 \tag{6.65}$$

$$\mu = \mu_1 = \mu_2 \tag{6.66}$$

$$D_{h1} = \frac{4A_{c1}}{P_{w1}} \tag{6.67}$$

$$D_{h2} = \frac{4A_{c2}}{P_{w2}} \tag{6.68}$$

$$D_{heq} = \frac{4(A_{c1} + A_{c2})}{(P_{w1} + P_{w2})} \tag{6.69}$$

Subsequently, with equivalent mechanical parameters defined for all variables in the electric circuit diagrams above, the equivalent viscous loss coefficient (S_{LAMeq}) for flow through the multipath cross-section of a partially-populated spacer segment is obtained by substituting Equation (6.55) and Equations (6.62) through (6.64) into Equation (6.53), which yields

$$\frac{\Delta P}{\frac{S_{LAM1} L\mu}{2{D_{h1}}^2 A_{c1}}} + \frac{\Delta P}{\frac{S_{LAM2} L\mu}{2{D_{h2}}^2 A_{c2}}} = \frac{\Delta P}{\frac{S_{LAMeq} L\mu}{2{D_{heq}}^2 (A_{c1}+A_{c2})}} \tag{6.70}$$

Rearranging and solving for (S_{LAMeq}), Equation (6.70) becomes

$$S_{LAMeq} = \frac{{D_{heq}}^2 (A_{c1} + A_{c2})}{\left(\frac{{D_{h1}}^2 A_{c1}}{S_{LAM1}} + \frac{{D_{h2}}^2 A_{c2}}{S_{LAM2}}\right)} \tag{6.71}$$

Equation (6.71) represents the final expression for estimating the equivalent viscous loss coefficient with the hydraulic diameters given by Equations (6.67) through (6.69) above. All parameters in these equations are directly or indirectly known based on the cross-sectional geometry of the spacer segment, except for the viscous loss coefficient of each individual flow branch. However, since each of these factors are only dependent upon its own singly-connected branch geometry, these values are readily determined/approximated from existing tabulated values for similar cross-sectional geometries available in published texts and related research articles. Relevant parameter values and calculation results for the equivalent viscous loss coefficient of the partially-populated spacer segment (from the equivalent assembly model) are provided in Table 6.7 below, including comparisons with the viscous loss coefficients from unheated and heated flow trials in Section 5.3 as a means of assessing the solution accuracy.

Table 6.7: Reference parameter values and calculations for the viscous loss coefficient of the partially-populated spacer segment using the proposed equivalent resistance method with comparisons to CFD simulation results.

EQUIVALENT RESISTANCE METHOD - VISCOUS LOSS COEFFICIENT FOR PARTIALLY-POPULATED SPACER SEGMENT FROM EQUIVALENT ASSEMBLY MODEL

Note: For ease of calculations, geometry values provided and utilized in the calculations below are based on the entire partially-populated spacer cross-section (*i.e.* values for each flow branch include consideration for all 688 symmetry segments).

[Flow Branch 1 Geometry]

Parameter:	Value:
P_w (m)	2.99445090
A_c (m^2)	4.885644E-03
D_h (m)	0.00652626

[Flow Branch 2 Geometry]

Parameter:	Value:
P_w (m)	6.19488489
A_c (m^2)	3.365737E-03
D_h (m)	0.00217324

[Equivalent Geometry]

Parameter:	Value:
$P_{w\,[1+2]}$ (m)	9.18933578
$A_{c\,[1+2]}$ (m^2)	8.251381E-03
D_{heq} (m)	0.00359172

[Estimated Viscous Loss Coefficient Per Flow Branch - Using Tabulated Reference Values Published]

Flow Branch:	~ X-Section Shape:	~ Aspect Ratio ():	Reference:	~ S_{LAM} ():
1	Parallel Plates	458.8 → Infinite	Incropera & DeWitt (2002)	96.0
2	Rectangular Tube	1.457	Incropera & DeWitt (2002)	59.0

[Final Equivalent Resistance Solution]

Parameter:	Value:
S_{LAMeq} ()	43.678796

[Details For Viscous Loss Coefficient Comparisons]

Analysis Description:	Section Discussed:	Viscous Loss Coefficient Parameter:
Proposed Equivalent Resistance Method	6.4.3	S_{LAMeq}
CFD Unheated, Pressure-Driven Flow Trials	5.3.2	$S_{LAMpressure}$
CFD Heated, Buoyancy-Driven Flow Trials	5.3.3	$S_{LAMbuoyancy}$

[Viscous Loss Coefficient Comparison Results]

Comparison #1	$S_{LAMpressure}$ ():	S_{LAMeq} ():	Difference (%):
	43.432205	43.678796	0.5678
Comparison #2	$S_{LAMbuoyancy}$ ():	S_{LAMeq} ():	Difference (%):
	43.359972	43.678796	0.7353

The example calculation above provides sufficient clarification of the procedure followed in order to obtain the best-estimate value in sensitivity study #4 for the unknown viscous loss coefficient of the partially-populated spacer segments. A comparison of the results in Table 6.7 illustrates a remarkable agreement between the calculated viscous loss coefficient using the proposed method and both the unheated and heated viscous loss coefficients determined from the CFD simulation solutions (within a 1% margin of error). The margin of error is also slightly smaller for the comparison between the proposed method and unheated CFD simulation values as expected, but the increased difference for the heated viscous loss coefficient is practically negligible. This lends significant supporting evidence towards the validity of the equivalent resistance method proposed.

As a further application example and evaluation of the proposed equivalent resistance method for estimating the viscous loss coefficients of flows through multipath cross-section segments, consideration is subsequently provided for the substantially more complicated cross-sectional geometry associated with an actual fully-populated spacer segment containing numerous branches of flow. The specific geometry addressed pertains to CAD solid model dimensions at the midplane of the fully-populated spacer segment utilized in the FLUENT CFD simulations from Section 5.2 for the single fully-populated fuel bundle run and spacer model. A comparison between the viscous loss coefficient from the CFD simulation trials in Section 5.2 and the equivalent resistance calculation is implemented in assessing the accuracy of the solution obtained using the proposed method. Details concerning this application example and evaluation comparison with the CFD simulations are briefly summarized below.

Details of Fully-Populated Spacer Evaluation:

- Unless otherwise noted, similar assumptions apply to the following calculations as given in the aforementioned example for the partially-populated spacer segment of the equivalent assembly model.
- Derivation of the equivalent (S_{LAM}) or (S_{LAMeq}) factor for cross-sections consisting of three or more singly-connected flow paths between common nodes is exactly analogous to the previous derivation for the two-path cross-section of the partially-populated spacer in the equivalent fuel assembly model. The formulation is easily expanded to incorporate any number of singly-connected flow paths working under the same stated assumptions.
- The only modification required to account for additional singly-connected flow branches traversing a common node junction (*i.e.* upstream inlet to downstream outlet of spacer segment) is the inclusion of a flow rate expression for each additional flow branch, which is added to the same side of the equality as existing flow rate expressions in the above two-branch example. This insures the mass flow rate is conserved across any common node connecting the cross-section flow branches.
- As a result of this understanding, a more appropriate general form of the equivalent (S_{LAM}) factor governing equation that is applicable to the current analysis or any other multipath flow system operating under the originally stated assumptions is given by

$$S_{LAM(eq)} = \frac{{D_{h(eq)}}^2 A_{c(total)}}{\sum_{n=1}^{n=m} \frac{{D_{h(n)}}^2 A_{c(n)}}{S_{LAM(n)}}} \tag{6.72}$$

with the hydraulic diameters defined as

$$D_{h(eq)} = \frac{4A_{c(total)}}{P_{w(total)}} \tag{6.73}$$

$$D_{h(n)} = \frac{4A_{c(n)}}{P_{w(n)}} \tag{6.74}$$

and the following subscript notation applied

(n) = a singly-connected branch; numbered between a common node as 1, 2…, m

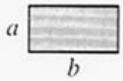
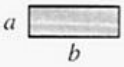
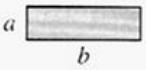
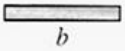
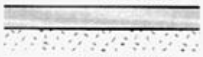
(m) = total number of singly-connected branches between a common node

(*total*) = sum of parameter value for all common node flow branches

(*eq*) = parameter value for single, equivalent, condensed path of all branches

- The (S_{LAM}) factor for each singly-connected flow branch is approximated from a list of tabulated values concerning the most similar cross-sectional geometries, which are based on data from a widely available, published text concerning general heat and mass transfer (Incropera and DeWitt, 2002). These values and cross-sectional shapes are illustrated in Figure 6.21 below.

[S_{LAM} Factor Reference Table; Adapted From (Incropera and DeWitt, 2002)]

Cross Section	$\frac{b}{a}$	$Nu_D \equiv \frac{hD_h}{k}$ (Uniform q_s'')	$Nu_D \equiv \frac{hD_h}{k}$ (Uniform T_s)	$f\,Re_{D_h}$ [= S_{LAM}]
	—	4.36	3.66	64
	1.0	3.61	2.98	57
	1.43	3.73	3.08	59
	2.0	4.12	3.39	62
	3.0	4.79	3.96	69
	4.0	5.33	4.44	73
	8.0	6.49	5.60	82
 Heated	∞	8.23	7.54	96
 Insulated	∞	5.39	4.86	96
	—	3.11	2.47	53

NOTE: A curve-fit to the above data for rectangular tube cross-sections is created to determine S_{LAM} factor values at intermediate aspect ratios using the TableCurve 2D software.

Figure 6.21: Reference Nusselt numbers and friction factors for fully-developed laminar flow through ducts of varying cross-sectional shape; adapted from (Incropera and DeWitt, 2002).

Therefore, with the details listed above and reference values approximated from Figure 6.21, all the equations and parameters necessary to complete the calculations outlined in the proposed equivalent resistance method are known. The analysis of

individual flow branches present in the fully-populated spacer cross-section requires examination of only three small subsections of the entire midplane geometry, which possesses ten unique flow branch shapes. These three groups of unique flow branch geometries are shown in Figure 6.22 below along with a descriptive overview of initial tasks in the calculation procedure. Furthermore, subsequent Figures 6.23 through 6.25 illustrate a more-detailed consideration of the flow branches in each group and the respective approximation values for the viscous loss coefficients from the reference table in Figure 6.21 above.

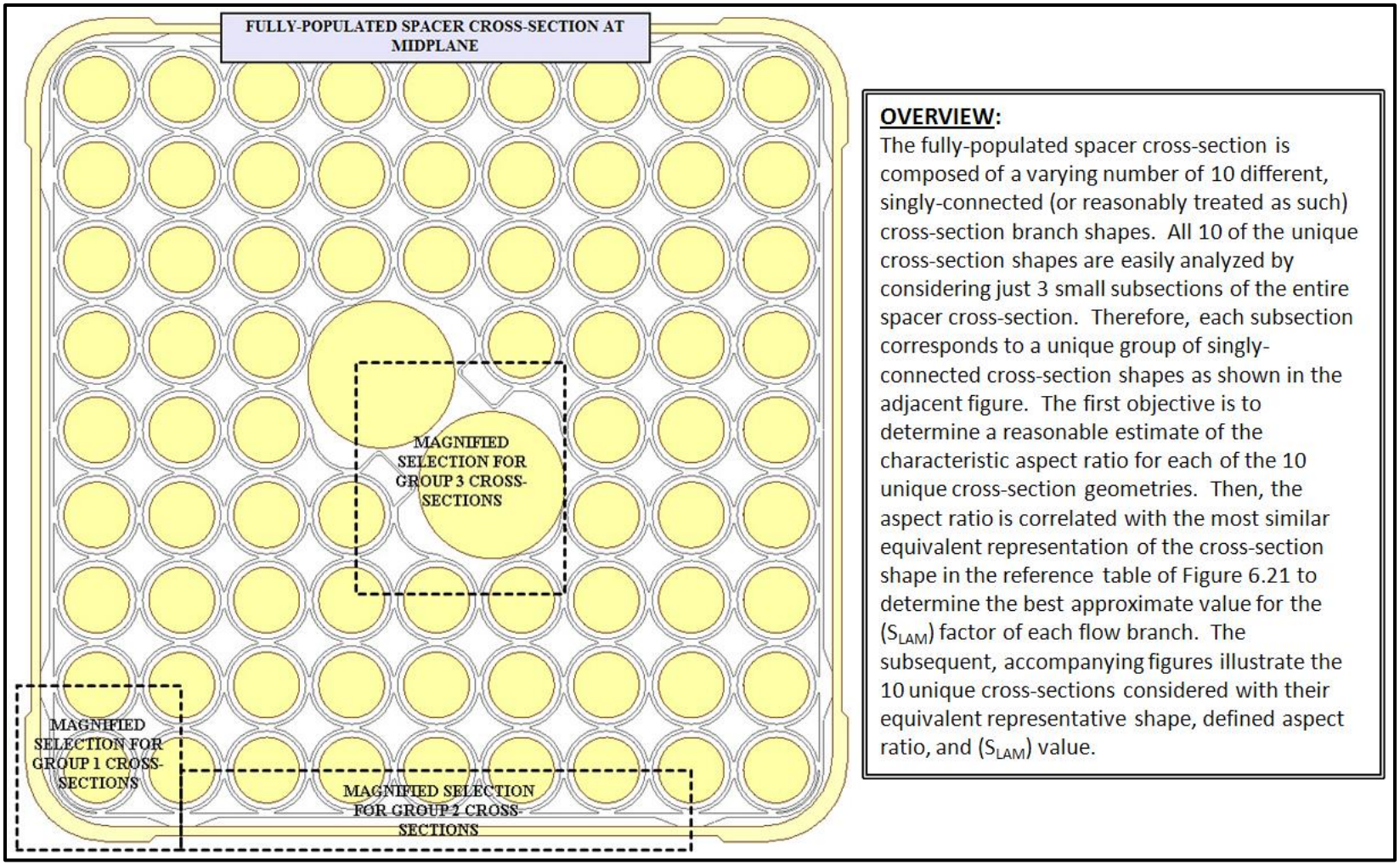
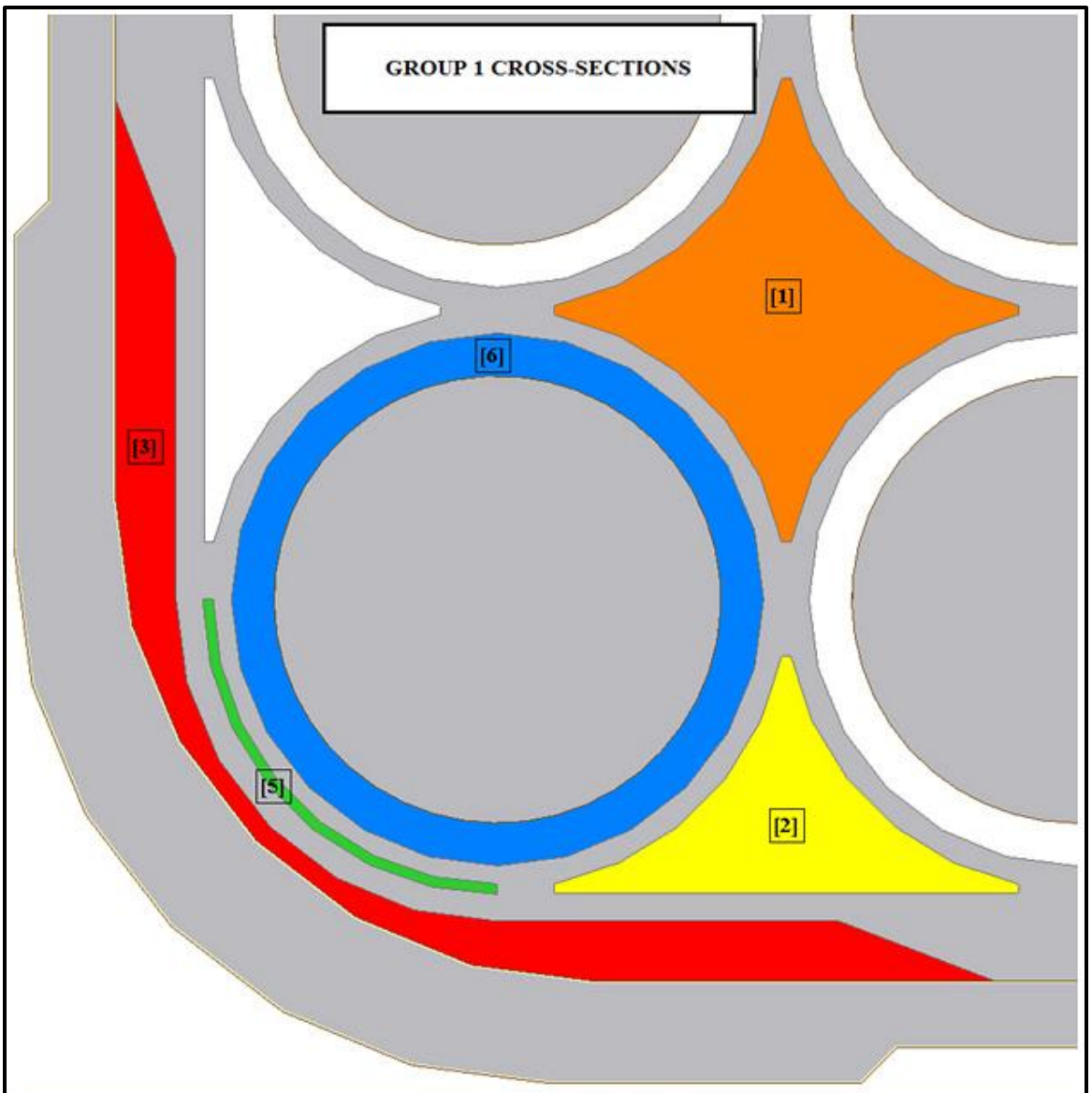
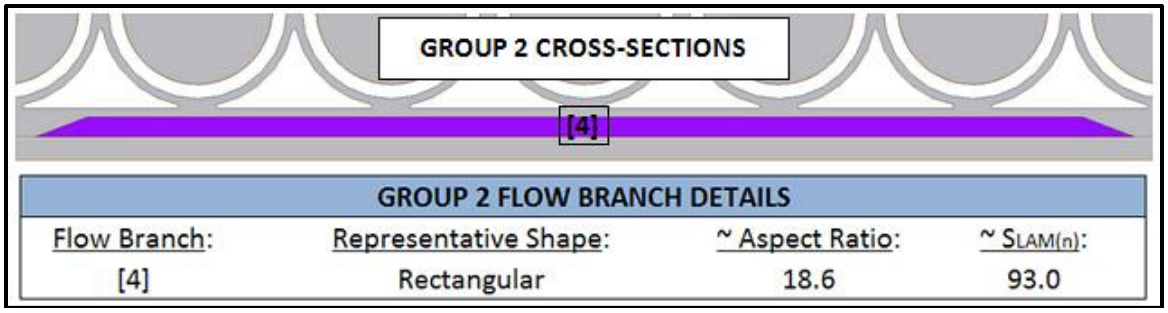


Figure 6.22: Overview of initial calculation considerations with designated locations of the three subsection groups.



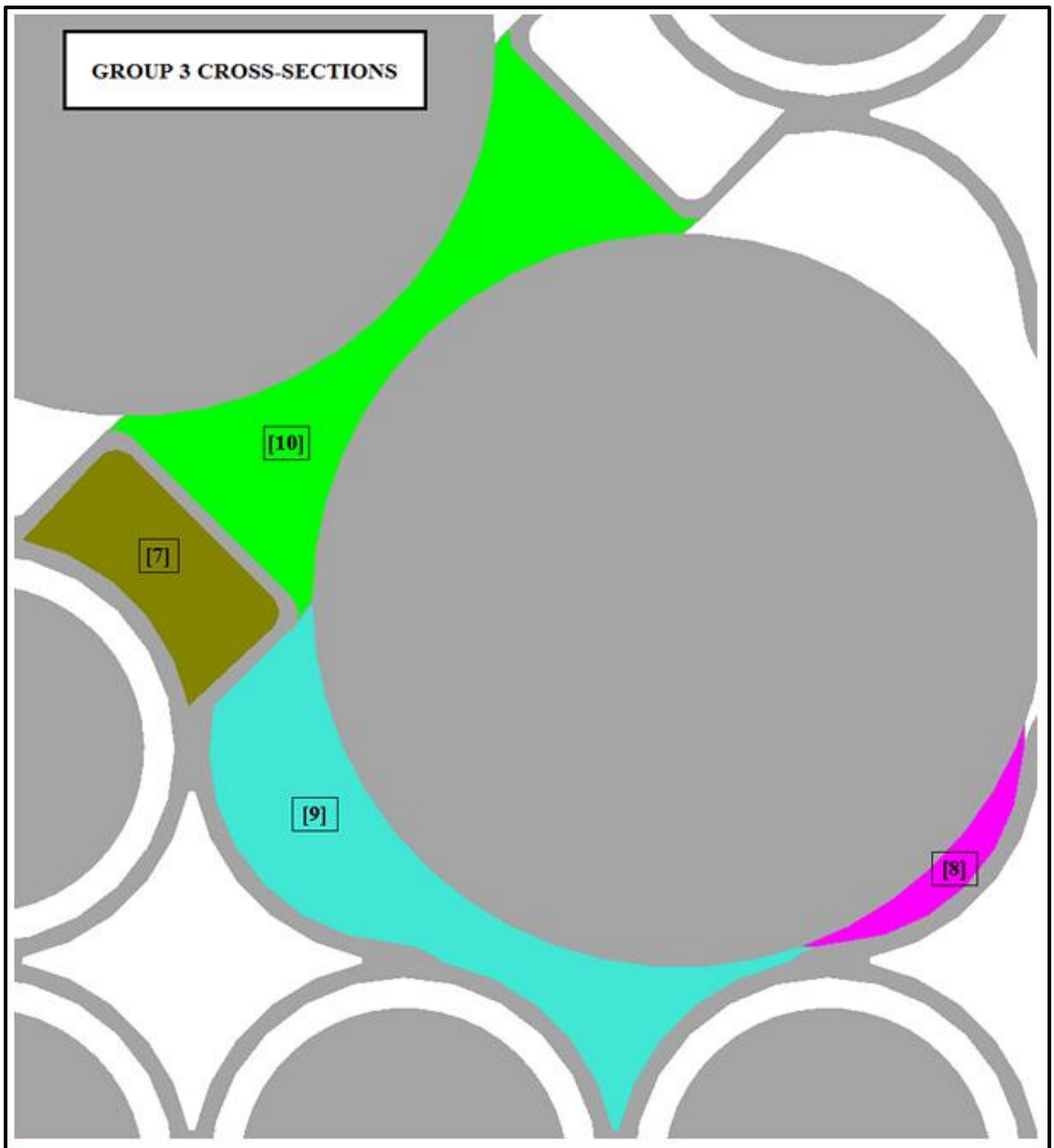
GROUP 1 FLOW BRANCH DETAILS			
Flow Branch:	Representative Shape:	~ Aspect Ratio:	~ $S_{LAM(n)}$:
[1]	Rectangular	1.0	57.0
[2]	Triangular	1.0	53.0
[3]	Rectangular	7.3	81.0
[5]	Parallel Plates	Infinite	96.0
[6]	Parallel Plates	Infinite	96.0

Figure 6.23: Enlarged view with details of unique flow branches in group 1 subsection.



GROUP 2 FLOW BRANCH DETAILS			
Flow Branch:	Representative Shape:	~ Aspect Ratio:	~ $S_{LAM(n)}$:
[4]	Rectangular	18.6	93.0

Figure 6.24: Enlarged view with details of unique flow branch in group 2 subsection.



GROUP 3 FLOW BRANCH DETAILS			
Flow Branch:	Representative Shape:	~ Aspect Ratio:	~ $S_{LAM(n)}$:
[7]	Rectangular	1.8	61.0
[8]	Rectangular	17.0	92.0
[9]	Rectangular	3.0	69.0
[10]	Rectangular	2.2	64.0

Figure 6.25: Enlarged view with details of unique flow branches in group 3 subsection.

Using approximated values for the friction factors (viscous loss coefficients) of unique flow branches from Figures 6.23 through 6.25 by substituting for the (S_{LAM}) flow branch parameters in Equation (6.72) above, a final estimate of the overall equivalent friction factor [*i.e.* $S_{LAM(eq)}$] for the entire multipath cross-section associated with fully-populated spacer segments is obtained according to the proposed method. Obviously, a number of geometry dimensions and intermediate calculations are necessary for each of the unique flow branch paths in order to acquire the remaining parameter values in Equation (6.72). These dimensions and the results from intermediate calculations are provided in Table 6.8 below, which also includes the final ascertained solution for the equivalent friction factor [$S_{LAM(eq)}$] and a comparison with the resolved value from Section 5.2 for congruent CFD simulations performed.

Table 6.8: Reference geometry and flow branch parameter calculations for estimating the viscous loss coefficient of a fully-populated spacer segment (based on afore detailed CAD solid model of assembly) using the proposed equivalent resistance method. Solution is compared with the result from CFD simulations outlined in Section 5.2 above.

EQUIVALENT RESISTANCE CALCULATION FOR $S_{LAM(eq)}$ IN FULLY-POPULATED SPACER SEGMENT BASED ON ASSEMBLY GEOMETRY FROM SOLID CAD MODEL

Flow Branch Reference Geometry And Intermediate Calculation Results - Base Unit Of Measurement Is Centimeters Due To Minute Length Scale

Unique Flow Branch:	No. Identical Branches:	$A_{c(n)}$ [cm^2]:	$P_{w(n)}$ [cm]:	$D_{h(n)}$ [cm]:	$\sum A_{c(n)}$ [cm^2]:	$\sum P_{w(n)}$ [cm]:	Approximate Shape:	~ a [cm]:	~ b [cm]:	~ Ratio:	~ $S_{LAM(n)}$:	$\sum D_{h(n)}^2 * A_{c(n)} / S_{LAM(n)}$ [cm^4]:
[1]	56	0.430030	3.485049	0.493571	24.08168	195.16272	Rectangular	1.74252	1.74252	1.0	57.0	0.10292283
[2]	32	0.228393	2.920337	0.312831	7.308581	93.450788	Triangular	0.57732	0.58891	1.0	53.0	0.01349514
[3]	4	0.407311	7.660197	0.212690	1.629245	30.640789	Rectangular	0.46663	3.41225	7.3	81.0	0.00090990
[4]	4	1.168500	16.41570	0.284727	4.674000	65.662813	Rectangular	0.83570	15.5800	18.6	93.0	0.00407440
[5]	4	0.029653	2.315418	0.051226	0.118611	9.261670	Parallel Plates	0.05241	2.26300	Infinite	96.0	0.00000324
[6]	74	0.409431	7.652920	0.214000	30.29791	566.31606	Parallel Plates	0.21400	7.65292	Infinite	96.0	0.01445337
[7]	2	0.313291	2.544525	0.492495	0.626582	5.089049	Rectangular	0.89850	1.64602	1.8	61.0	0.00249145
[8]	2	0.094861	2.022520	0.187609	0.189721	4.045041	Rectangular	0.13024	2.21499	17.0	92.0	0.00007258
[9]	4	0.746835	6.005773	0.497411	2.987339	24.02309	Rectangular	1.54245	4.66697	3.0	69.0	0.01071191
[10]	1	0.761220	6.012365	0.506436	0.761220	6.012365	Rectangular	1.00734	2.21367	2.2	64.0	0.00305056

Equivalent Branch Values And Resulting $S_{LAM(eq)}$ Solution

Flow Branch:	$A_{c(total)}$ [cm^2]:	$P_{w(total)}$ [cm]:	$D_{h(eq)}$ [cm]:	$S_{LAM(eq)}$:
Equivalent	72.674887	999.664381	0.290797	40.382364

S_{LAM} Calculation Comparison - Equivalent Resistance Method And CFD Simulations (Section 5.2)

Equivalent Resistance Calculation:	FLUENT CFD Simulations (Section 5.2):	Difference [%]:
40.382364	38.503605	4.8794

Again, a comparison of viscous loss coefficients in Table 6.8 reveals a remarkable agreement between the determined value from CFD simulation trials in Section 5.2 and the estimated result from equivalent resistance calculations introduced in the preceding discussion. Despite multiple differences in the two domains evaluated (re-emphasized in concluding remarks), the equivalent friction factor [$S_{LAM(eq)}$] estimate for the multipath cross-section characterizing a fully-populated spacer segment falls within a 5% margin of error with these CFD computations. This incurs a number of reasonably justifiable inferences with regard to the validity of this proposed method as well as several unexplained observations concerning the CFD simulation results in Section 5.2 above. However, to avoid an impetuous statement of apparent findings, it is necessary to establish a level of confidence in the acquired solution first. In particular, the sensitivity of expected variances in parameters identified for a few unique flow branches (*i.e.* representative shape, aspect ratio, and tabulated friction factor) on the final calculated result certainly warrants an adequate assessment.

The initial task in outlining an appropriate preliminary sensitivity study for the equivalent [$S_{LAM(eq)}$] factor analysis above is identifying the scope of consideration in which the greatest amount of meaningful information is permissibly extracted from a limited number of simple computations. Obviously, the expected degree of variance in parameters associated with flow branch cross-sections [1], [2], [4], [5], [6], and [7] (see Figures 6.23 through 6.25) is minimal since these singly-connected paths are well represented by corresponding equivalent shapes from the reference tabulations in Figure 6.21 for several basic geometrical cross-sections. The aspect ratios of these flow branches are largely consistent along edge lengths (or a given edge perspective) and fall within a low (< 2) or high (> 18) range of values, or correlating assumption directly inherent to a property of the shape. This significantly simplifies the process of accurately

selecting the (S_{LAM}) factor associated with the aspect ratio of each flow branch cross-section from the reference tabulations in Figure 6.21 above. As a result, the range of subjectivity/degree of variance in the appropriate characteristic and (S_{LAM}) factor values selected for branch cross-sections [1], [2], [4], [5], [6], and [7] is justifiably limited and further examination of the solution sensitivity with regard to parameter variances in these cross-sections is not considered or warranted in a preliminary sensitivity study.

On the other hand, remaining unique branch cross-sections [3], [8], [9], and [10] possess plausible arguments of a larger discernible range for arbitrary selection and assessment of the reasonably equivalent shapes opted for in assigning the viscous loss coefficients of these branches from the (S_{LAM}) factor reference table. In addition, the highly curved surface extents associated with these cross-sections encompass aspect ratios that vary continuously along the bounding edge lengths, which are certainly open to interpretation in terms of the characteristic aspect ratio defined for each branch path. Therefore, the range of subjectivity permissible in the determination of parameter values and corresponding (S_{LAM}) factors for these branch cross-sections necessitates the quantification of influences on the final solution from potential variances in order to assess the accuracy, reliability, and practicality of employing the proposed method for analyzing this multipath flow system. Accordingly, the scope of the sensitivity study is focused on determining the response of the equivalent [$S_{LAM(eq)}$] factor solution to variances in the interpretation of characteristic parameter values associated with flow branch cross-sections [3], [8], [9], and [10].

The most efficient procedure for accomplishing this objective involves direct modification of the (S_{LAM}) factor value associated with each branch cross-section. This process is parallel to the evaluation of numerous permissible interpretations for potential representation of a branch cross-section pertaining to the equivalent shape selection,

definition of cross-section aspect ratio, or other variation leading to similar modification of the (S_{LAM}) value. A reasonable approximation of the max deviation for (S_{LAM}) likely to exist between two different interpretations of the same flow branch cross-section is 20% of the value identified in the original calculation above. Furthermore, maximum (S_{LAM}) value deviations represented by 20% increases over the original branch values correspond to equivalent [$S_{LAM(eq)}$] outcomes of greatest percent increase in comparative difference with the CFD simulation solution from Section 5.2 since equivalent resistance method calculations in Table 6.8 slightly over predict the viscous loss factor estimate. Therefore, the max deviations to original (S_{LAM}) factor values coincide with 20% increases in the original flow branch values, which maximize potential difference with the CFD simulation solution for sensitivity evaluations of worst-case assessments.

In summary, this sensitivity study evaluates the individual contribution of branch cross-sections [3], [8], [9], and [10] on the equivalent [$S_{LAM(eq)}$] factor solution resulting from a 20% increase in the original (S_{LAM}) value selected for each flow branch, which is compared with the original calculation and FLUENT CFD model results. Finally, the effect of increasing the (S_{LAM}) factor value of all branch cross-sections ([3], [8], [9], and [10]) by 20% at the same time is considered and the resulting solution difference is examined. Sensitivity study calculations and results are provided in Table 6.9 below.

Table 6.9: Sensitivity study results of equivalent resistance calculation from Table 6.8 above for the viscous loss coefficient of a fully-populated spacer segment based on the CAD solid model geometry of the fuel assembly. Results are compared with original calculation and FLUENT CFD model solutions.

SENSITIVITY STUDY OF EQUIVALENT RESISTANCE CALCULATION ABOVE FOR $S_{LAM(eq)}$ IN A FULLY-POPULATED SPACER SEGMENT

Original $S_{LAM(eq)}$ Values - Equivalent Resistance Method And CFD Simulations (Section 5.2)

Equivalent Resistance Calculation:	FLUENT CFD Simulations (Section 5.2):	Difference [%]:
40.382364	38.503605	4.8794

Sensitivity Study Results and Comparisons

Flow Branch:	Original $S_{LAM(n)}$:	Original $S_{LAM(n)}$ + 20%:	New $S_{LAM(eq)}$:	Difference With Original [%]:	Difference With CFD Trials [%]:
[3]	81.0	97.2	40.422644	0.0997	4.9841
[8]	92.0	110.4	40.385574	0.0079	4.8878
[9]	69.0	82.8	40.861721	1.1870	6.1244
[10]	64.0	76.8	40.517727	0.3352	5.2310
[3], [8], [9], [10]	Above Values	Above Values	41.045162	1.6413	6.6008

The sensitivity study results in Table 6.9 for the original calculation defined by the equivalent resistance method clearly illustrate an almost negligible influence on the final [$S_{LAM(eq)}$] solution arising from variances in the interpretations of flow branches [3], [8], [9], and [10]. Despite a 20% increase in the original (S_{LAM}) factor values associated with these branch cross-sections, the equivalent viscous loss coefficient remains within 2% of the originally calculated value and 7% of the FLUENT CFD simulations. Consequently, the solution in Table 6.8 for the original calculation is reasonably concluded as an accurate representation of the actual value obtained based on the proposed method and establishes a sufficient level of confidence in the acquired solution.

In comparing the equivalent resistance method and CFD simulation results, it is important to remember the two fundamental discrepancies between each respective formulation for the viscous loss coefficient. First, as outlined in Section 5.2, the resulting [$S_{LAM(eq)}$] factor from CFD simulations is based on differential pressures between bounding planes located 0.001 cm up- and downstream of the spacer since interior boundaries do not exist in the FLUENT model at the exact bundle-spacer interface extents due to the assembly method required in creating the final large mesh file. Second, although both solutions employ the hydraulic diameter value associated with the midlength cross-sectional geometry of the fully-populated spacer segment, the actual hydraulic diameter of the flow domain represented in the FLUENT CFD model varies across the spacer length due to the outer lobes. Hence, the larger hydraulic diameter for the true flow domain modeled in the FLUENT CFD simulations leads to the expectation of smaller viscous pressure losses in comparison to the equivalent resistance calculation, which assumes the midlength cross-sectional geometry of the spacer across its entire length. This is supported by the results illustrated in Tables 6.8 and 6.9 in which the

value of [$S_{LAM(eq)}$] predicted by the equivalent resistance method is larger than the coefficient value calculated from the FLUENT CFD simulations.

The analyses above provide a significant degree of merit with regard to the accuracy of the proposed equivalent resistance method and the documented results substantiate a number of important findings. Foremost, correctness of the constructed mesh models, FLUENT simulation setups, convergence criteria, and resulting evaluations of solution data for the CFD trials discussed in Sections 5.2 and 5.3 are indirectly supported by the relative agreement in [$S_{LAM(eq)}$] values presented for comparison. In addition, the similarity further suggests that differences between SNL experimental data and FLUENT CFD simulations highlighted in Section 5.2 for airflows traversing a fully-populated bundle run and spacer segment are likely due to exclusion of the spacer mixing devices/structures within the flow domain model as previously indicated. Nevertheless, these inferences are of particular interest to the overall understanding of simulated airflows through a typical fuel assembly and certainly warrant further investigation.

6.4.4 Critical Heat Input Rate Trials

The primary function of the MATLAB program is to predict the buoyancy-induced mass flow rate and temperature response of a prototypic fuel assembly under complete LOCA conditions. Exploiting this capability in a preliminary assessment of the experimental fuel assembly at SNL, utilizing explicit information provided in their concluding hydraulic analysis report (Durbin and Lindgren, 2005), is the intended focus of the following efforts that represent the final phase of research conducted. The validation studies outlined in Sections 6.4.1 and 6.4.2 clearly demonstrate the ability of the program code to produce accurate solutions to such problems (given reasonable

estimates of the required input parameters) through detailed comparisons with benchmark CFD simulations.

Although inputs for fuel assembly dimensions are directly acquired from the hydraulic analysis report by SNL (Durbin and Lindgren, 2005), the stated loss coefficients derived from experimental measurements pertain to unheated (pressure-driven) flows, the values for component stratification terms do not exist, and initial guesstimates for component temperatures as well as the solution control values remain undetermined. As a result, a rationale is adopted in order to address these issues, which is briefly summarized according to the following key points.

- Despite inarguable differences between the simulated flow domain in the equivalent assembly model and the actual flow domain of the experimental fuel assembly at SNL, the observed pressure loss response from both the SNL experiments and CFD simulation trials using the equivalent assembly model are relatively similar as previously discussed in Section 5.3 above. Therefore, it is reasonably plausible to assume that any general trends discerned within the solution responses stemming from analyses employing the equivalent assembly model are also likely to exist as part of the solution responses associated with the actual flow domain of the experimental fuel assembly.
- Building on this assumption, the sensitivity trials in Section 6.4.2 indicate a minimal influence of the stratification term values on the solution as long as a linear temperature distribution (*i.e.* $S = 0.5$) is assumed for each component flow segment. A similar result is shown between calculations implementing loss coefficients derived from both unheated and heated airflows, which include the assumption of a linear temperature distribution in the unheated computation

comparisons as well. Consequently, the use of stratification term values correlating with a linear temperature distribution for each component flow segment as well as loss coefficients from unheated airflow trials are expected to produce reasonably accurate solutions that represent the actual response of the SNL experimental fuel assembly.

- In addition, the sensitivity trials in Section 6.4.2 illustrate a nearly negligible influence of initial temperature estimates and solution control inputs on the resulting solution provided the built-in defaults (or close to) are selected and air property temperatures as well as fuel heat input rates are within the approximate range of initially examined base values. Thus, default (or near) values/options are implemented within acceptable range for the net fuel heat input rate and solutions are expected to remain unaffected by these designated initial inputs.

The MATLAB program computations are performed under the above-stated assumptions with consideration of the built-in defaults for other initial inputs described previously in this chapter. Obviously, the objective is to employ the MATLAB program in an effort to provide the best preliminary prediction of the buoyancy-induced mass flow rate and temperature response within the SNL experimental fuel assembly using the loss coefficients and dimensions identified in their final report of the experimental hydraulic analysis (Durbin and Lindgren, 2005). This assessment includes an examination of net fuel heat input rates spanning from 300 to 1650 watts with component lengths determined by the average connecting pressure port distances and models the fuel assembly structure between Ports 2 and 17 as shown in Figure 3.1 above; the upper and lower tie plates are not considered in these calculations. Based on the justifications given for required parameter inputs, the results are expected to have a reasonable level of numerical

significance toward the true representation of an experimental fuel assembly heat up with corresponding steady-state solutions within a 20% margin of anticipated error. The essential input values for the MATLAB program parameters are listed in reference Table 6.10 below and the graphical results for critical variables of interest (at fuel heat input rates designated, if a selection is warranted) are illustrated in Figures 6.26 through 6.31, which follow.

Table 6.10: Reference input values for the critical heat rate trials based on reported data from SNL pressure loss experiments, which are required by the MATLAB program code.

MATLAB PROGRAM - CRITICAL HEAT INPUT RATE TRIALS - ESSENTIAL REFERENCE VALUES FROM SNL EXPERIMENTS AND INITIAL ESTIMATES

[Geometry]	Value:
Bundle Run Length (m)	0.47
Grid Spacer Length (m)	0.0445
Total Model Length (m)	4.0715
Fully-Populated Bundle Flow Area (m^2)	0.00979
Fully-Populated Spacer Flow Area (m^2)	0.00737
Partially-Populated Bundle Flow Area (m^2)	0.01060
Partially-Populated Spacer Flow Area (m^2)	0.00835

[Geometry - Continued]	Value:
Fully-Populated Bundle Hydraulic Diameter (m)	0.01190
Fully-Populated Spacer Hydraulic Diameter (m)	0.00289
Partially-Populated Bundle Hydraulic Diameter (m)	0.01410
Partially-Populated Spacer Hydraulic Diameter (m)	0.00357

[Ambient Environment Conditions]	Value:
Pressure (Pa)	82833.7622
Temperature (K)	294.4034
Gravitational Acceleration (m/s^2)	9.81

Hydraulic Loss Coefficients - Reported By SNL - Determined From Unheated Pressure Loss Experiments Of Fuel Assembly

[Viscous Terms - S_{LAM}]	Value:
Fully-Populated Inlet Bundle Run ()	77.0
Fully-Populated Bundle Run ()	77.0
Fully-Populated Grid Spacer ()	32.0
Partially-Populated Bundle Run ()	60.0
Partially-Populated Grid Spacer ()	21.0

[Form Terms - k]	Value:
Fully-Populated Inlet Bundle Run ()	0.84
Fully-Populated Bundle Run ()	0.84
Fully-Populated Grid Spacer ()	1.80
Partially-Populated Bundle Run ()	0.88
Partially-Populated Grid Spacer ()	1.90

[Other Remaining Inputs]

Note: Values are specified in prior discussions of the assumptions implemented and the built-in defaults utilized.

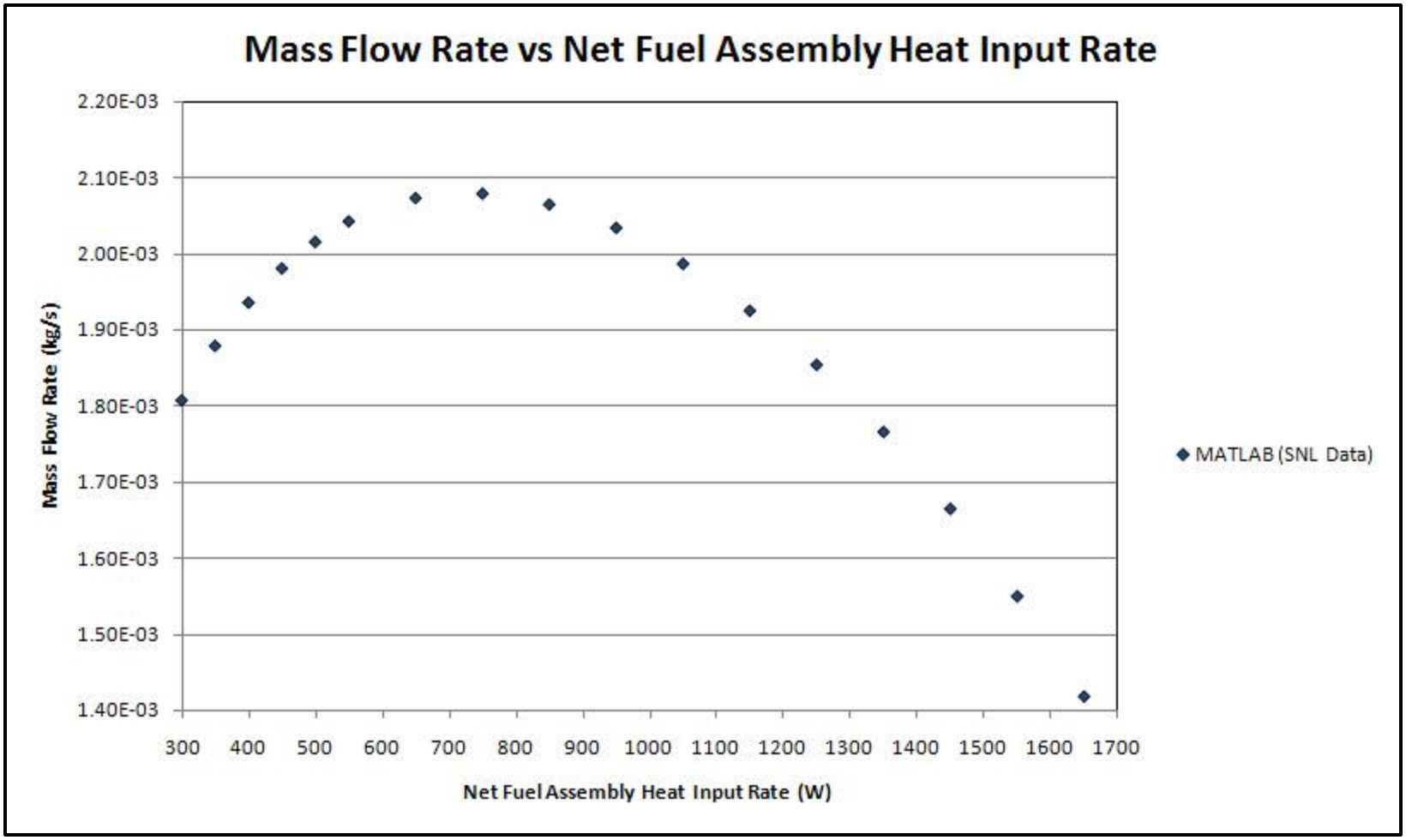


Figure 6.26: Critical heat input rate trial results from MATLAB program for mass flow rate versus net heat input rate using SNL experimental data based on the hydraulic analysis report by Durbin and Lindgren (2005).

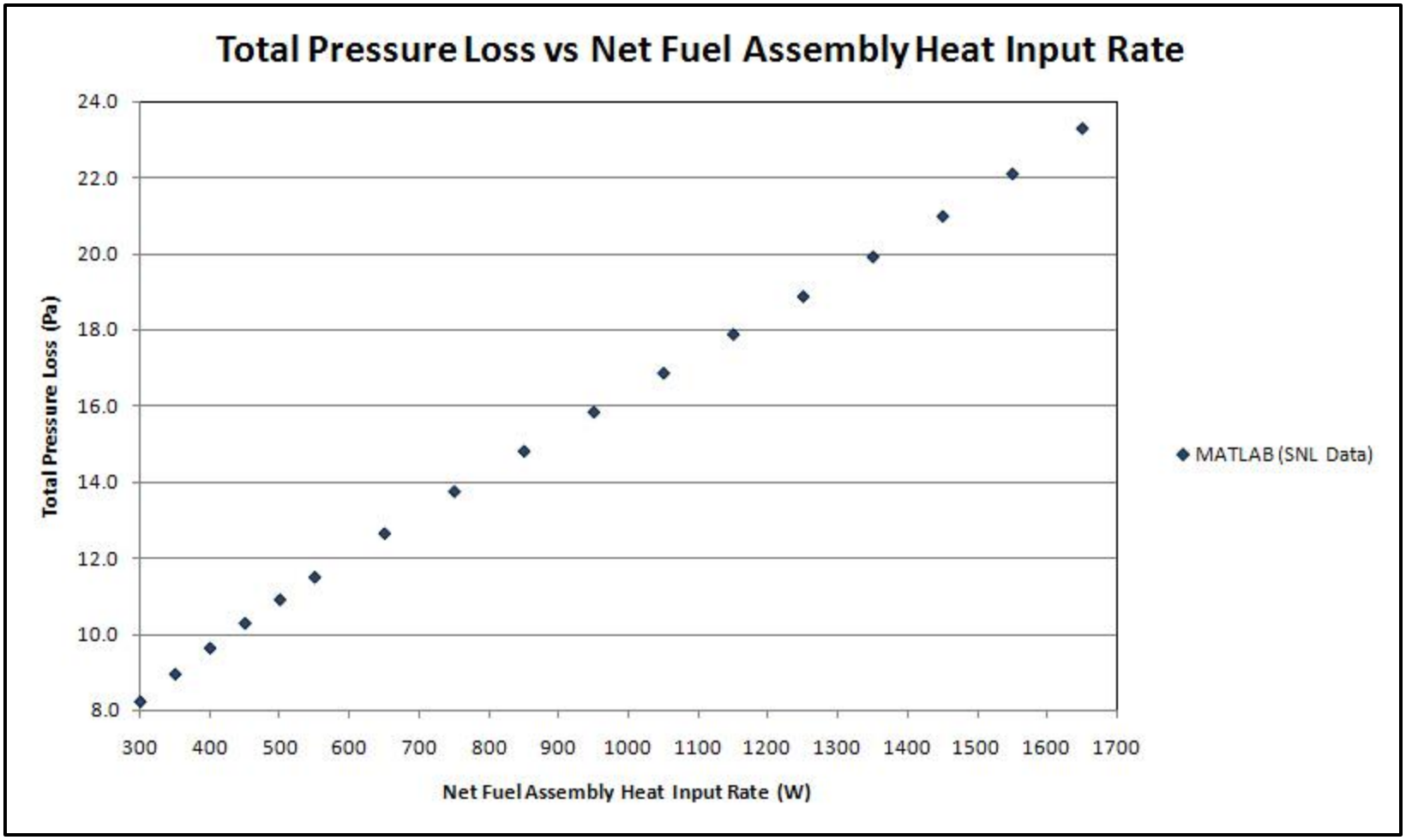


Figure 6.27: Critical heat input rate trial results from MATLAB program for total pressure loss versus net heat input rate using SNL experimental data based on the hydraulic analysis report by Durbin and Lindgren (2005).

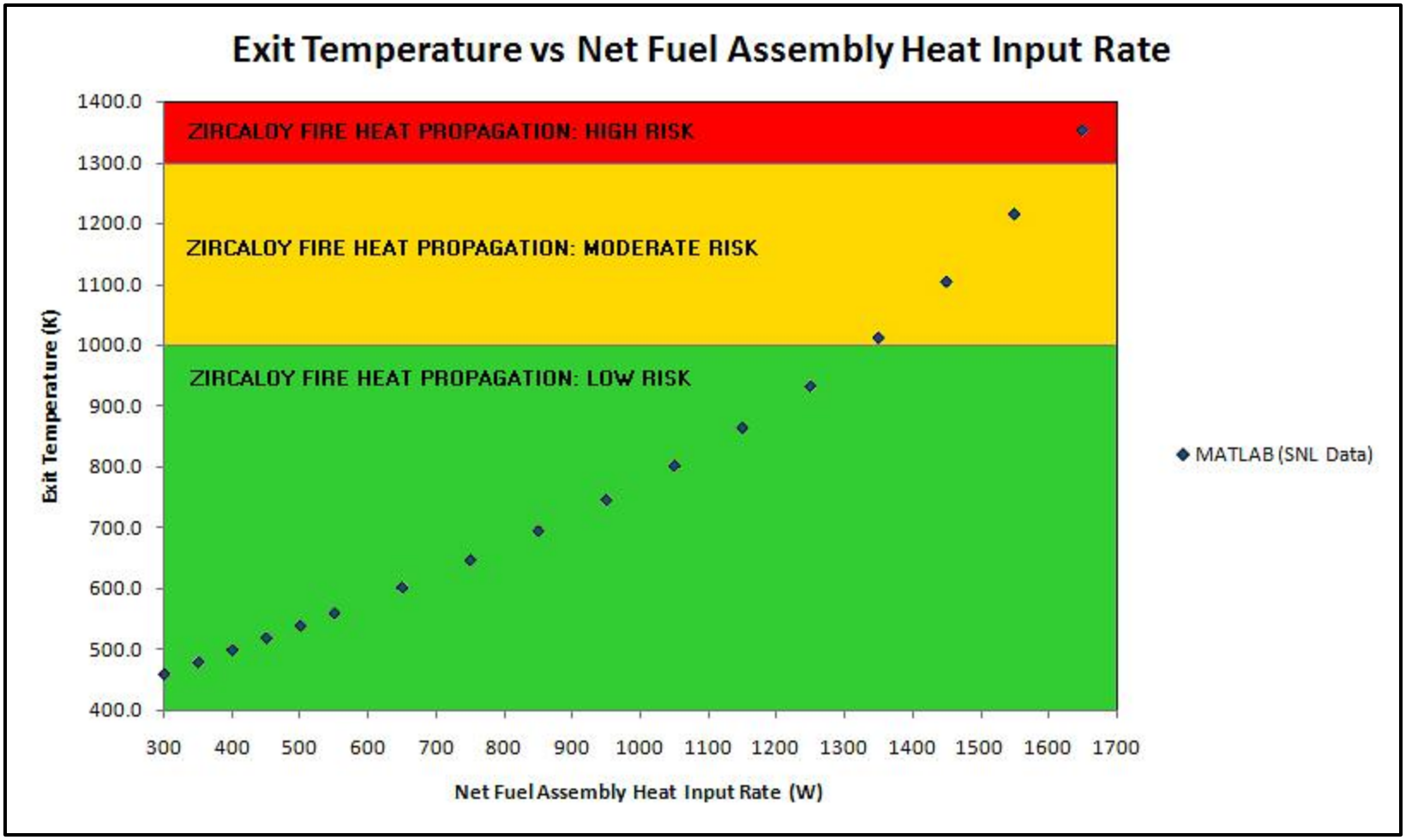


Figure 6.28: Critical heat input rate trial results from MATLAB program for assembly exit temperature versus net heat input rate using SNL experimental data based on the hydraulic analysis report by Durbin and Lindgren (2005).

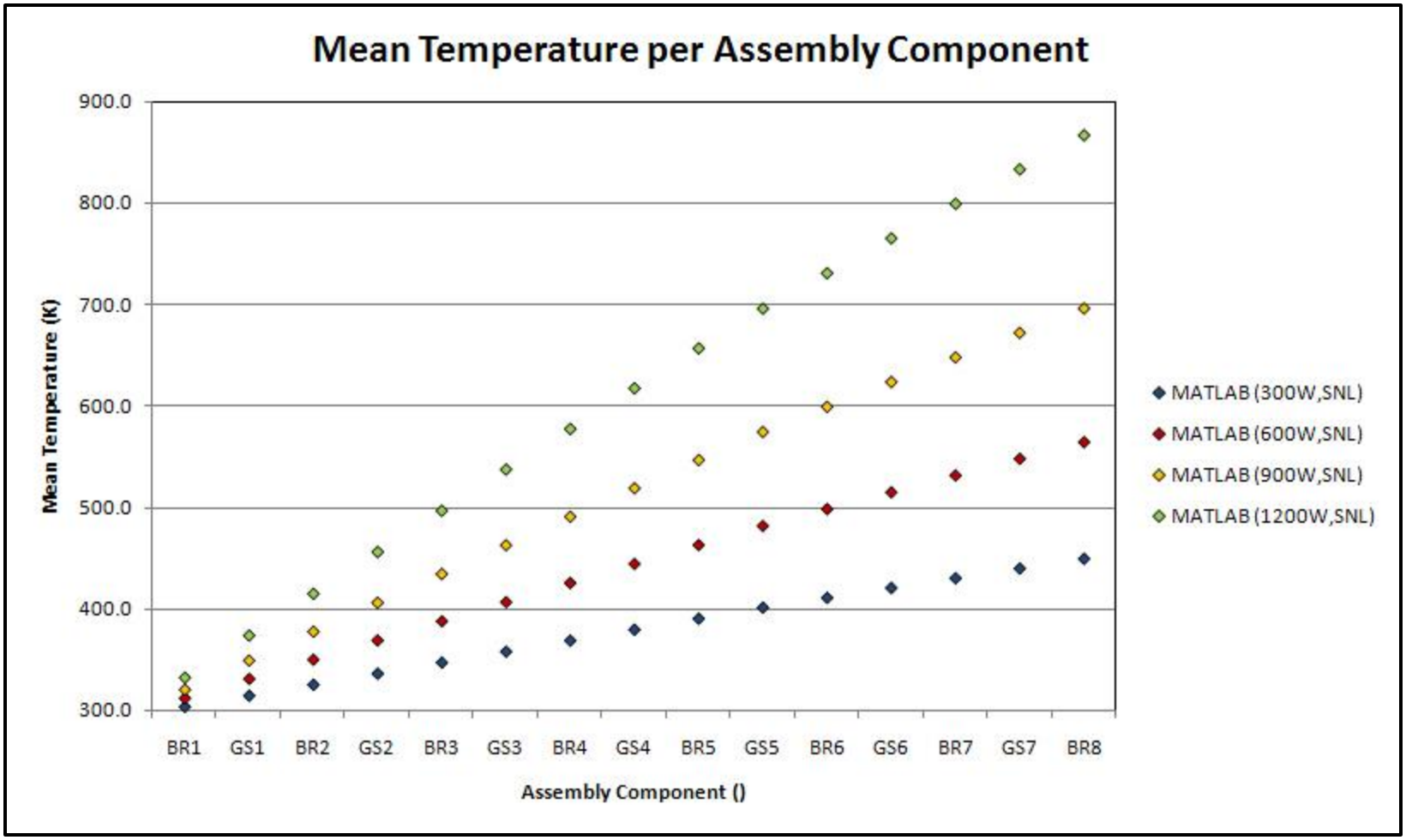


Figure 6.29: Critical heat input rate trial results from MATLAB program for mean temperature per assembly component (at select heat rates) using SNL experimental data based on the hydraulic analysis by Durbin and Lindgren (2005).

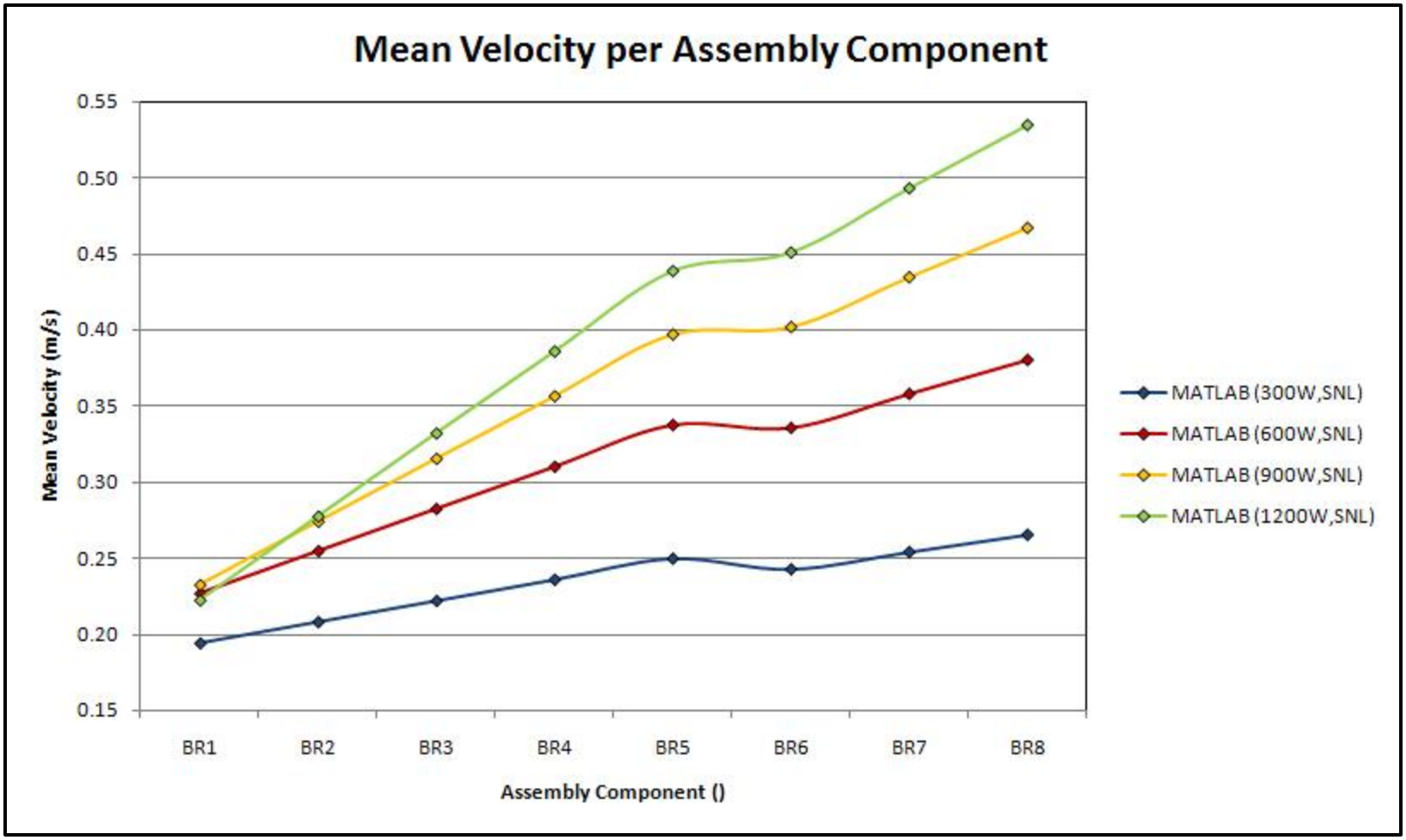


Figure 6.30: Critical heat input rate trial results from MATLAB program for mean velocity per bundle run component (at select heat rates) using SNL experimental data based on the hydraulic analysis by Durbin and Lindgren (2005).

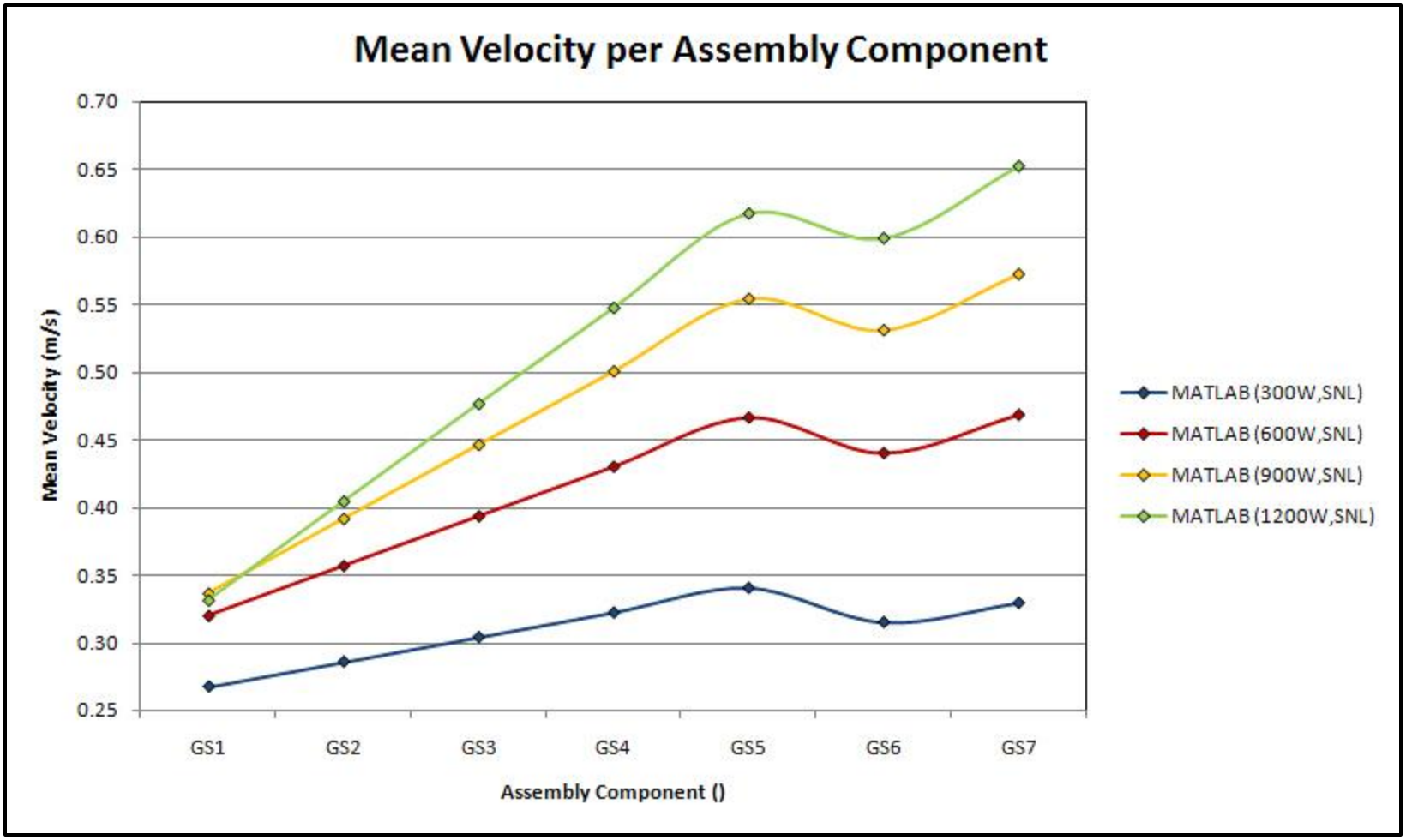


Figure 6.31: Critical heat input rate trial results from MATLAB program for mean velocity per spacer component (at select heat rates) using SNL experimental data based on the hydraulic analysis by Durbin and Lindgren (2005).

The graphical results from the critical heat input rate trials displayed in Figures 6.26 through 6.31 provide a valuable reference set of parameter values that afford SNL with a conservative assessment of the predicted system response pertaining to their specific experimental fuel assembly and reported measurement data. This information is intended to serve as an additional reference to considerations regarding the planning of future research initiatives in this area. In particular, the exit temperature versus net heat input rate response shown in Figure 6.28 exemplifies the relevance of program output toward the evaluation of risk associated with rate of heat removal requirements for decaying spent fuel. The plot area in this figure is divided into three risk levels delineated by temperature lines at 1000 and 1300 Kelvin representing the approximate temperatures upon which initial and rapid exothermic air oxidation of the zircaloy fuel cladding is traditionally observed, respectively (Lindgren, 2004).

This process significantly increases the heat up rate of fuel as well as heat removal requirements of buoyancy-induced airflow through the assembly, which poses a greater risk for the propagation of a severe SFP accident scenario. According to Figure 6.28, the MATLAB program predicts the risk transitions for 1000 and 1300 Kelvin to occur near the net heat input rates of 1350 and 1600 watts, respectively. Although these critical heat rate trials are based on a conservative viewpoint and a number of limiting assumptions are incorporated within the calculated solutions, such results provide an invaluable understanding of the general flow field and thermal behavior anticipated at the larger scale with minimal attributed cost. This is especially important during initial research and development phases where efficiency and computational resource costs are critical components of any project endeavor. However, further evaluation of the program is strongly recommended prior to exclusive utilization in preliminary SFP accident

analyses and additional research efforts to refine as well as improve upon the physics modeled within the code are necessary.

6.5 FINAL REMARKS

In summary, this chapter introduces relevant theory and initial research undertaken in the development and validation of a buoyancy-driven flow program written in the MATLAB programming language, which imparts solution estimates for the buoyancy-induced mass flow rate and temperature response within the flow domain of a prototypic 9x9 BWR fuel assembly under complete LOCA conditions. The discussions include detailed explanations and examples of the derivations necessary to acquire the governing equations in their final form. A thorough description of the methodology and assumptions giving rise to the designated solution procedure along with inherent code structure is outlined in a manner consistent with future modifications conducive to the addition of supplemental subroutines expanding the scope of physics modeled. Nonetheless, general construction of the base code is briefly concluded with the development of a reference pseudocode skeleton representing the essential framework elements of the MATLAB program.

The validity, application range, and confidence level associated with the MATLAB code are established through numerous evaluations and sensitivity studies conducted across a broad range of considerations. Comparisons with benchmark solutions from FLUENT CFD simulations noted in Chapter 5 are utilized in assessing the numerical accuracy of results obtained for several basic case trials using the MATLAB code. A comparative analysis shows remarkable agreement between the predicted parameter values from both the CFD simulations and MATLAB program calculations.

The sensitivity studies further demonstrate a wide range of application circumstances whereby solutions attained from the program fall within an acceptable margin of error if reasonable estimates are retained (via any experimental measurements/approximation method/published data) for the key input parameters.

Other primary observations from the sensitivity studies include the following: (1) nearly congruent solution responses for trials implementing loss coefficients based on both heated (buoyancy-driven) airflows with specific stratification terms (measured) as well as unheated (pressure-driven) airflows with constant stratification terms (assuming linear temperature distribution) and (2) exploiting the former with remarkably observed agreement between CFD simulation and MATLAB program results provides support (though not definitive) for the principle application of simple constitutive equations to model the hydraulic losses of complex assembly airflows with reasonable accuracy. In other words, these two observations support (although not definitively) the premise that constitutive/phenomenological equations representing the hydraulic flow losses within a fuel assembly (as implemented in the MELCOR SFP model and current MATLAB program) are capable of capturing this aspect of the flow-field solution with acceptable accuracy provided loss coefficients are determined correctly. Moreover, the observations suggest that loss coefficients derived from pressure measurement data associated with the unheated (pressure-driven) airflow experiments at SNL in conjunction with a constant value for stratification terms assuming a linear temperature distribution, are likely sufficient for calculating accurate solutions of the internal assembly flow field.

In an attempt to address the uniquely difficult situation of approximating the viscous loss coefficient without specific experimental reference data for a component cross-section comprised of multiple flow branch segments (*i.e.* fuel assembly designs containing spacers), a simple method based on the analogous evaluation of parallel-

connected resistors around a closed circuit loop is proposed for estimating such viscous loss coefficients. A comparative analysis of the viscous loss coefficients calculated using the proposed equivalent resistance method and those derived from the CFD simulations employing the equivalent assembly as well as fuel bundle run and spacer model reveals that solutions obtained according to the proposed method are completely consistent with the simulation results. This congruency between the FLUENT CFD simulations and equivalent resistance method (indeed all aforementioned comparison agreements as well) affords an increased assurance with respect to correctness of the constructed mesh models, FLUENT simulation setups, convergence criteria, and resulting evaluations of solution data for CFD trials discussed in Sections 5.2 and 5.3 above. The similarity further implies that differences between SNL experimental data and FLUENT CFD simulations highlighted in Section 5.2 for airflows traversing a fully-populated bundle run and spacer segment are likely due to exclusion of the spacer mixing devices/structures within the flow domain model as formerly indicated.

Exercising a number of presumed assumptions founded on the cumulative results as well as observations gained through preceding program assessment trials and validation studies, a best estimate analysis of the buoyancy-induced mass flow rate and temperature response of the SNL experimental fuel assembly is conducted using data available from the SNL hydraulic analysis report (Durbin and Lindgren, 2005). The solution information of foremost importance from the MATLAB program in these critical heat input rate computations is the extraction of approximate fuel decay heat limits (each corresponding to an anticipated observation temperature) for reaching both the initial and rapid exothermic air oxidation temperature points of the fuel cladding. For observed initial and rapid exothermic air oxidation temperatures of 1000 and 1300 Kelvin, the fuel

decay heat limits predicted by the MATLAB program are 1350 and 1600 watts, respectively.

On a final note, caution is strongly emphasized with regard to the implementation of this buoyancy-driven flow program as a primary/sole reference analysis tool, especially for modeling conditions outside the basic physics or assembly flow domain geometries addressed in the comparative analyses above. Although preliminary validation studies and solution comparisons with FLUENT CFD simulations are extremely promising at every stage of assessment undertaken thus far, the substantiation of confidence in solutions beyond the narrow range of computational applications benchmarked at present necessitates continued research efforts to refine as well as improve upon the physics modeled within the code itself. Unfortunately, the scarcity of benchmark solution data from related experimental research/endeavors constituted a previous hindrance to the depth as well as efficiency of benchmark comparative analyses performed up until the latest program modifications.

However, as clearly demonstrated in almost every prior research effort relating to the MATLAB program, a necessary capability or notable benefit toward development and/or validation of the code is afforded by the FLUENT CFD modeling software, which is expected to serve as an invaluable reference tool for continued efforts to enhance and upgrade the program. The importance is particularly augmented under situations in which experimental research data is completely unavailable and exemplifies one of the many indirect benefits of employing the FLUENT CFD modeling software aside from a full-scale numerical simulation of the SNL experimental fuel assembly, which still requires further model development and accuracy assessment before truly considered as a plausible option.

Chapter 7: Conclusion

7.1 REVIEW

Prompted by a number of uncertainties concerning the physical basis and interpretation of implied assumptions for variances in related behavior-defining parameters, an extensive endeavor was undertaken by Sandia National Laboratories to ascertain a more definitive understanding as well as accurate characterization of resultant physical flow-field phenomena and governing fuel assembly responses in a SFP complete LOCA scenario. Ordained as a primary assessment tool in modeling such postulated accident scenarios, the principal objective of SNL root efforts is to effect notable improvements in both the predicted responses and parameter characterization uncertainties in the MELCOR SFP model code. As such, a need exists to provide basic thermal hydraulic data in accordance with a SFP complete loss-of-coolant accident to enable precise calibration of requisite input parameters, enhance and/or facilitate code validation, and reduce reservations associated with interpretation of analysis results concerning the MELCOR SFP model (and potentially physical experiments).

In an attempt to fulfill the above-stated need, Sandia National Laboratories employed a joint computational and experimental effort to provide essential data and analysis for the improvement and validation of the MELCOR SFP model. This is the fundamental motivation behind larger investigational directives from which current dissertation research focus and objectives are derived.

Accordingly, the impetus behind this computational fluid dynamics modeling and numerical experimentation was to provide additional support and understanding of observed fluid flow behavior as well as the physical mechanisms that govern typical fuel

assembly thermal hydraulic response under SFP complete LOCA conditions through the application of commercial FLUENT CFD software. The capability of FLUENT CFD models to afford supplementary high-fidelity solutions and characteristic hydraulic flow-field data supporting the calibration of accurate parameter values for the MELCOR SFP model in addition to physical measurements obtained through SNL experimental trials was of specific focus to the underlying objectives of this computational assessment. Functionality of the CFD software and accuracy of model developments were largely discerned through comparative analyses with measured experimental values acquired at Sandia National Laboratories.

In order to accomplish the outlined project goals in an organized approach, key areas of investigational focus were individually addressed through several different phases of research undertaken in this work. Foremost, these included computational analysis efforts using commercial FLUENT CFD software to develop models for simulations involving both the direct replication of SNL fuel assembly experiments and indirect assessment of prototypic fuel assembly responses in complete LOCA scenarios as well as the formulation of an original simplified program code for implementation in supplementary analysis applications. The principal tasks of initial consideration among computational endeavors were concerned with the direct replication of fuel assembly flow-field segments under SNL experimental conditions for comparative analysis, which subsequently required the greatest dedication of project scheduling cost and high performance computing resources across the board.

Therefore, to promote rapid and accurate construction of flow domains for mesh generation among CFD models created, a complete computer-aided design (CAD) solid model of Sandia National Laboratories' prototypic Global Nuclear Fuel (GNF) 9x9 BWR (GE 11 BWR/2-3) experimental fuel assembly was constructed using SolidWorks

commercial 3D CAD software. Fuel assembly components within this model are precisely based on dimensioned part drawings from the manufacturer where available and on-hand measurements from accessible segments of the actual SNL experimental apparatus for any remaining dimensions necessary. The only intentional simplifications included in this solid model construction are present in the limited representation of the mixing device structures on assembly spacer components. This CAD solid model development also serves as an invaluable reference and widely accessible resource to SNL researchers since final high-temperature experiments proposed are expected to result in complete destruction of the prototypic experimental assembly.

As noted, the first two phases of computational research concentrated on the potential utilization of CFD modeling and simulation for direct analysis of the resultant flow-field responses within a prototypic SFP fuel assembly under complete LOCA conditions similarly stipulated for SNL experimental trials. However, due to the absence of former supporting attempts to implement commercial CFD software in a reasonably related manner for modeling a 9x9 BWR fuel assembly with such detailed geometrical precision under the postulated accident conditions of interest, this work started at the ground level building towards the progression of larger scale domain modeling. Therefore, this research is largely directed toward the need to provide a complete preliminary assessment of computational costs, benefits, model development considerations, solution uncertainties, simulation pitfalls, subsequent large-scale model obstacles, solution accuracy, and the potential application of results associated with the employment of FLUENT CFD software to analyze such accident flow domains.

The computational models developed for the first phase of research included a fully-populated fuel bundle run between two grid spacer components. These initial models were setup to perform base case simulations and parameter studies of various

mass flow rates within a theorized range of temperature induced buoyancy-driven airflow expected to develop through single-span fuel bundle segments. The analyses offered a detailed cross-examination for a multitude of resulting parametric values that are excellent for comparative evaluation. This included preliminary scoping runs gauging resource requirements, grid-independence, periodicity and symmetry applicability, validity of viscous model options and wall treatment method (if one is needed), appropriateness of boundary conditions, and other pertinent setup characteristics which establish a sufficient level of accuracy and confidence in the final model implementation and solution(s) obtained. These models evaluated flow rates ranging from 200 to 600 slpm assuming unheated/incompressible/constant property airflow.

The high-fidelity computational model developed for the second phase of research was a single span fully-populated bundle run and spacer segment within the prototypic fuel assembly structure. Grid resolution and key setup characteristics were based on the results and knowledge gained from the first phase of research completed. This model included over 260 million-volume cells and required 800 GB of aggregate memory as well as 400 processing cores to execute on the Ranger Sun Constellation Linux Cluster. Approximately 2,000 to 3,500 iterations were required to reach convergence. The second phase model produced numerical results for static and total pressures across the system as well as high-fidelity graphical profiles for pertinent flow-field parameters. These computational results were generally comparable to Sandia National Laboratories' experimental measurements. Differences were concluded to arise largely as a consequence of simplifications implemented with regard to spacer mixing devices represented within the FLUENT CFD model construct.

The final phase of CFD modeling and simulation included the development and application of a simplified hydraulic-scale equivalent model of the complete SNL

experimental fuel assembly structure. Results stemming from numerical experiments utilizing this equivalent fuel assembly model clearly demonstrated a flow-field response range within 10% of SNL experimental measurements for unheated, forced airflow trials. Furthermore, the comparison between the unheated and heated fuel assembly simulations show substantial agreement among the hydraulic loss coefficient values obtained from each. System properties were acquired from these models in a form conducive to integration into a more computationally efficient, simplified model development.

The concluding stage of this research effort included the development of a single loop force balance model within MATLAB to assess buoyancy-driven flows in a SFP complete loss-of-coolant accident scenario. Again, system properties and parameter values for this model were obtained from the final phase of CFD modeling and simulation, which utilized the equivalent fuel assembly model. In short, the corresponding comparative analysis of results exploiting the MATLAB buoyancy-driven flow program supports the conclusion that hydraulic loss coefficients derived from unheated, forced and heated, buoyancy-driven flows may be reasonably interchanged, which was shown to remain accurate to within 5% of the actual heated flow-field benchmark. Ultimately, this MATLAB buoyancy-driven flow model/program code produced flow-field results in agreement with the equivalent fuel assembly model from the final phase of CFD modeling and simulation trials to within 1% and 3% of all compared flow-field values.

7.2 Recommendations and Implications for Future Work

From a CFD perspective, minor simplifications in fuel assembly geometry and thus flow domain structure had a significant influence on the pressure loss response as

well as overall flow-field development. For example, the mixing devices on spacer components within the 9x9 BWR fuel assembly must be modeled with geometry as close to an exact representation as possible. However, exact modeling is made difficult due to the extent of curvilinear surface profiles defining the spacer segments and it is not clear what degree of simplification is acceptable to maintain a reasonable margin of error, which must be balanced against increases in model development cost and computational resource requirements. As such, further investigations are necessary in order to clarify this complexity and ensure that mixing devices on BWR fuel assembly spacers are properly modeled and ultimately reflected within the solution values of resolved flow field domains.

The high-fidelity models utilized in this research required a significant investment of FLUENT serial and parallel computing licenses and necessitated a substantial dedication of high performance computing and visualization system resources. Although these models constitute several of the highest resolution (largest volume-cell count) geometric mesh model representations of BWR fuel assembly segments known in open science, they are far from representing a complete BWR fuel assembly structure, which provides a weak scale for estimating the enormity of remaining development effort and additional computational resources necessary to achieve a full-scale model. On the other hand, the FLUENT CFD models and software repeatedly demonstrated the potential to sufficiently resolve the majority of anticipated phenomena from the fluid flow spectrum expected to develop under SFP complete LOCA scenarios and function as useful indirect analysis and development tools.

Combined with the understanding that developments of larger-scale models will become more practical as computational capabilities improve over the next 5 to 10 years, the continued progression toward larger-scale simulations of fuel assembly responses

under SFP accident conditions is strongly supported using the accomplishments and shortcomings revealed throughout these computational research developments for guiding and directing future CFD modeling work. Additional efforts by other researchers to employ CFD modeling software in the evaluation of fuel assemblies under other SFP complete LOCA conditions/scenarios or even flow geometry is of utmost benefit towards establishing more accurate and efficient CFD model developments for SFP LOCA analysis and highly endorsed.

Finally, the simplified and computationally efficient MATLAB buoyancy-driven flow program (and code design) developed for this work predicted and captured an accurate as well as precise representation of the pertinent thermal hydraulic parameters and steady-state response of a prototypic 9x9 BWR fuel assembly encompassed in a SFP system under complete LOCA conditions for specified net heat input rate values. Consequently, the incorporation of such simplified approaches employing the use of constitutive relationships accompanied with well-founded reference parameter data for complex physical phenomena into nuclear reactor accident analysis codes such as MELCOR for modeling complex physical phenomena is encouraged. These simplified numerical methods, when benchmarked properly, are able to produce results of near-equivalent accuracy to large, complex 3D CFD models without a fraction of the computational expense.

Appendix A: Extended MATLAB Program Documentation

A.1 Base Model Code with Comments

The underlying code development for the MATLAB buoyancy-driven flow program is structured as a compilation of three distinct MATLAB M-files, each performing a specific set of functions that are essential to the overall computational solution routine. In particular, control of the basic input/output tasks as well as iteration process and function calls are handled by the script M-file labeled 'buoyancy.m', which serves as the execution backbone of the entire program. Numerical values for the mass flow rate are calculated across iteration loops by the function M-file labeled 'massflowbuoyancy.m', which uses the built-in MATLAB function roots(p) to find zeros of the governing polynomial equation. A second function M-file labeled 'propertybuoyancy.m' is implemented with the purpose of calculating air property values at designated temperatures using the same expressions introduced in Section 5.3 for FLUENT CFD simulations involving heated assembly airflows. The developed code for each program M-file with descriptive comments is included below.

< BUOYANCY.M >

```
%M-file script called buoyancy.m which calls M-file function massflowbuoyancy.m
%for approximating (Mdot) zero(s) of the governing function.  The outer loop of
%this M-File script successively updates the mean air properties of the flow
%and the inner loop solves for the convergence of mean temperatures (from which
%the air properties are derived by calling M-file function propertybuoyancy.m).
%Using an energy balance over the control volume 9x9 BWR fuel assembly, a
%buoyancy-induced mass flow rate solution is obtained.  This model references the
%BWR fuel assembly components by three region types: fully-populated inlet bundle
%region (termed inlet) if necessary, fully-populated bundle and spacer region
%(termed full or fully-populated), and partially-populated bundle and spacer
%region (termed partial or partially-populated).  Therefore, enter all parameter
%values according to this description (if applicable to the designated parameter
%name listed in the program inputs below).

%Clear command window, workspace variable values, and functions from memory.

clc;
clear all;
clear functions;

%Declare static global variables needed within any functions or sub-functions
%called by this script.

global Sbr1;
global Sgs1;
global Sbr2;
global Sgs2;
global Sbr3;
global Sgs3;
global Sbr4;
global Sgs4;
global Sbr5;
global Sgs5;
global Sbr6;
global Sgs6;
global Sbr7;
global Sgs7;
global Sbr8;
global g;
global Dbrf;
global Dbrp;
global Dgsf;
global Dgsp;
global Lbr;
global Lgs;
global Acbrf;
global Acbrp;
global Acgsf;
global Acgsp;
global Slaminlet;
global Slambrf;
global Slamgsf;
global Slambrp;
global Slamgsp;
global Kinlet;
global Kbrf;
global Kgsf;
global Kbrp;
global Kgsp;
global Tin;
```

```
global Patm;
global Rgas;
global Qbrf;
global Qgsf;
global Qbrp;
global Qgsp;
global rho_not;

%Declare varying global variables needed within any functions or sub-functions
%called by this script.

global Tbr1m;
global Tgs1m;
global Tbr2m;
global Tgs2m;
global Tbr3m;
global Tgs3m;
global Tbr4m;
global Tgs4m;
global Tbr5m;
global Tgs5m;
global Tbr6m;
global Tgs6m;
global Tbr7m;
global Tgs7m;
global Tbr8m;
global rho_br1m;
global Cp_br1m;
global mu_br1m;
global nu_br1m;
global rho_gs1m;
global Cp_gs1m;
global mu_gs1m;
global nu_gs1m;
global rho_br2m;
global Cp_br2m;
global mu_br2m;
global nu_br2m;
global rho_gs2m;
global Cp_gs2m;
global mu_gs2m;
global nu_gs2m;
global rho_br3m;
global Cp_br3m;
global mu_br3m;
global nu_br3m;
global rho_gs3m;
global Cp_gs3m;
global mu_gs3m;
global nu_gs3m;
global rho_br4m;
global Cp_br4m;
global mu_br4m;
global nu_br4m;
global rho_gs4m;
global Cp_gs4m;
global mu_gs4m;
global nu_gs4m;
global rho_br5m;
global Cp_br5m;
```

```
global mu_br5m;
global nu_br5m;
global rho_gs5m;
global Cp_gs5m;
global mu_gs5m;
global nu_gs5m;
global rho_br6m;
global Cp_br6m;
global mu_br6m;
global nu_br6m;
global rho_gs6m;
global Cp_gs6m;
global mu_gs6m;
global nu_gs6m;
global rho_br7m;
global Cp_br7m;
global mu_br7m;
global nu_br7m;
global rho_gs7m;
global Cp_gs7m;
global mu_gs7m;
global nu_gs7m;
global rho_br8m;
global Cp_br8m;
global mu_br8m;
global nu_br8m;
global Mdot_initial;
global Tmprop;

%Display program introduction and general instruction information.

disp(' ');
disp('------------------------------------------------------------------------');
disp('INTRODUCTION:  The modeled domain in this program is a prototypical   ');
disp('9x9 BWR fuel assembly in a SFP under a severe accident scenario in    ');
disp('which all coolant has been drained from the pool.  It is assumed that ');
disp('the net decay heat input rate is balanced at steady-state by the heat ');
disp('removal rate from the system due to a buoyancy-induced air flow rate  ');
disp('through the domain.  This program model uses an energy balance over   ');
disp('the control volume BWR fuel assembly to obtain a buoyancy-induced mass');
disp('flow rate solution.  This model references the BWR fuel assembly      ');
disp('components by three region types: fully-populated inlet bundle region ');
disp('(termed inlet) if necessary, fully-populated bundle and spacer region ');
disp('(termed full or fully-populated), and partially-populated bundle and  ');
disp('spacer region (termed partial or partially-populated).  Therefore,    ');
disp('enter all parameter values according to this description (if          ');
disp('applicable to the designated parameter name listed in the program     ');
disp('inputs below).                                                        ');
disp(' ');
disp('M-FILE DETAILS:  This program is run by the M-file script called      ');
disp('buoyancy.m, which calls M-file function massflowbuoyancy.m for        ');
disp('approximating (Mdot) zero(s) of the governing function.  The outer    ');
disp('loop of this M-File script successively updates the mean air          ');
disp('properties of the flow and the inner loop solves for the convergence  ');
disp('of mean temperatures (from which the air properties are derived by    ');
disp('calling M-file function propertybuoyancy.m).                          ');
disp('------------------------------------------------------------------------');
disp(' ');

%Define or prompt user for required parameter values, initial estimates, and
```

```
%approximation method criteria needed for determining a final mass flow rate
%solution.

fprintf ('\n');
disp    ('[Enter variable values and criteria required for solution procedure]  ');
disp    (' ');
disp    (' ');
disp    ('-------------------------------------------------------------------------');
disp    (' ');
disp    ('Prototypical 9x9 BWR fuel assembly geometric values exist as defaults ');
disp    ('for this program.  Use these default values?                          ');
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt1    = input ('\n Enter Selection: ');
if (opt1 == 2),
    fprintf ('\n');
    Dbrf  = input ('\n Enter full fuel bundle hydraulic diameter (Dbrf [m]): ');
    Dbrp  = input ('\n Enter partial fuel bundle hydraulic diameter (Dbrp [m]): ');
    Dgsf  = input ('\n Enter full fuel spacer hydraulic diameter (Dgsf [m]): ');
    Dgsp  = input ('\n Enter partial fuel spacer hydraulic diameter (Dgsp [m]): ');
    Lbr   = input ('\n Enter length of a single fuel bundle (Lbr [m]): ');
    Lgs   = input ('\n Enter length of a single fuel spacer (Lgs [m]): ');
    Acbrf = input ('\n Enter cross-sectional area of flow in full bundle (Acbrf [m^2]): ' ↙
);
    Acbrp = input ('\n Enter cross-sectional area of flow in partial bundle (Acbrp [m^2]) ↙
: ');
    Acgsf = input ('\n Enter cross-sectional area of flow in full spacer (Acgsf [m^2]): ' ↙
);
    Acgsp = input ('\n Enter cross-sectional area of flow in partial spacer (Acgsp [m^2]) ↙
: ');
else,
    Dbrf  = 0.011907211068;
    Dbrp  = 0.014067268950;
    Dgsf  = 0.002907971473;
    Dgsp  = 0.003591720194;
    Lbr   = 0.473330;
    Lgs   = 0.030480;
    Acbrf = 9.6882917747e-03;
    Acbrp = 1.0463838532e-02;
    Acgsf = 7.2674887567e-03;
    Acgsp = 8.2513807269e-03;
end;
disp    (' ');
disp    ('-------------------------------------------------------------------------');
disp    (' ');
disp    ('Ambient / inlet environment variable values upon which experimental   ');
disp    ('and computational experiments were conducted exist as defaults for    ');
disp    ('this program.  Use these default values?                              ');
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt2    = input ('\n Enter Selection: ');
if (opt2 == 2),
    fprintf ('\n');
    g       = input ('\n Enter gravitational acceleration (g [m/s^2]): ');
    Tin     = input ('\n Enter ambient / inlet air temperature (Tin [K]): ');
    Patm    = input ('\n Enter atmospheric pressure of ambient air (Patm [Pa]): ');
    Rgas    = input ('\n Enter Universal Gas Constant for air (Rgas [J/kg-K]): ');
    rho_not = Patm/(Rgas*Tin);
    %Density of inlet / ambient air (kg/m^3)
else,
```

```
    g       = 9.81;
    Tin     = 294.4034;
    Patm    = 82833.7622;
    Rgas    = 286.689655172414;
    rho_not = Patm/(Rgas*Tin);
end;
disp    (' ');
disp    ('------------------------------------------------------------------------');
Qt      = input ('\n Enter net fuel assembly heat input rate (Qt [W]): ');
disp    (' ');
disp    ('------------------------------------------------------------------------');
disp    (' ');
disp    ('The net heat input rate for each component in the full or partial     ');
disp    ('region is based on the total length of all fuel rods in a component   ');
disp    ('compared to the total length of all fuel rods in the associated region');
disp    ('in which the component resides.  However, the fraction of the net fuel');
disp    ('assembly heat input rate designated to the full or partial region is  ');
disp    ('adjustable if desired.  The fraction value, based on the total length ');
disp    ('of all fuel rods in a region compared to the total length of all fuel ');
disp    ('rods in the entire fuel assembly, exists as a default for each region ');
disp    ('in this program.  Use these default values?                           ');
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt3    = input ('\n Enter Selection: ');
if (opt3 == 2),
    fprintf ('\n');
    Qfractf = input ('\n Enter fraction of net fuel assembly heat input rate to fully-pop ↙
ulated region (Qfractf []): ');
    Qf      = (Qfractf*Qt);
    %Net heat input rate to entire fully-populated region (W)
    Qp      = (Qt - Qf);
    %Net heat input rate to entire partially-populated region (W)
    Qbrf    = (((5*Lbr)/(5*Lbr + 5*Lgs))*Qf)/5;
    %Heat input rate to a single full fuel bundle (W)
    Qgsf    = (Qf - 5*Qbrf)/5;
    %Heat input rate to a single full fuel spacer (W)
    Qbrp    = (((3*Lbr)/(3*Lbr + 2*Lgs))*Qp)/3;
    %Heat input rate to a single partial fuel bundle (W)
    Qgsp    = (Qp - 3*Qbrp)/2;
    %Heat input rate to a single partial fuel spacer (W)
else,
    Qfractf = (74*(5*Lbr + 5*Lgs))/((74*(5*Lbr + 5*Lgs)) + (66*(3*Lbr + 2*Lgs)));
    Qf      = (Qfractf*Qt);
    Qp      = (Qt - Qf);
    Qbrf    = (((5*Lbr)/(5*Lbr + 5*Lgs))*Qf)/5;
    Qgsf    = (Qf - 5*Qbrf)/5;
    Qbrp    = (((3*Lbr)/(3*Lbr + 2*Lgs))*Qp)/3;
    Qgsp    = (Qp - 3*Qbrp)/2;
end;
disp    (' ');
disp    ('------------------------------------------------------------------------');
disp    (' ');
disp    ('The viscous and form loss coefficients for each component as well as  ');
disp    ('losses due to flow development in the inlet region, which are based on');
disp    ('CFD computational experiments modeling buoyancy-induced flow in a     ');
disp    ('simplified model domain having similar geometric component length     ');
disp    ('scales as a prototypical 9x9 BWR fuel assembly, exist as defaults for ');
disp    ('this program.  The defaults may not accurately represent these losses ');
disp    ('in an actual 9x9 BWR fuel assembly, but do provide reasonable starting');
disp    ('estimates when lacking specific measurement data.  Caution should be  ');
```

```
disp    ('exercised if using these defaults outside the net fuel assembly heat  ');
disp    ('input range of 300 to 1200 Watts from which these default values are  ');
disp    ('based.  Use these default values?                                     ');
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt4    = input ('\n Enter Selection: ');
if (opt4 == 2),
    fprintf ('\n');
    Slaminlet = input ('\n Enter value of (f*Re) for fully developed flow through inlet f↙
uel bundle (Slaminlet []): ');
    Slambrf   = input ('\n Enter value of (f*Re) for fully developed flow through full fu↙
el bundles (Slambrf []): ');
    Slamgsf   = input ('\n Enter value of (f*Re) for fully developed flow through full fu↙
el spacers (Slamgsf []): ');
    Slambrp   = input ('\n Enter value of (f*Re) for fully developed flow through partial↙
 fuel bundles (Slambrp []): ');
    Slamgsp   = input ('\n Enter value of (f*Re) for fully developed flow through partial↙
 fuel spacers (Slamgsp []): ');
    f1        = input ('\n Enter enhancement due to flow development or pressure defect n↙
umber (f1 []): ');
    Kbrinlet  = input ('\n Enter form loss coefficient for inlet fuel bundle (Kbrinlet [] ↙
): ');
    Kinlet    = (Kbrinlet + f1);
    %Sum of inlet fuel bundle form loss and enhancement due to flow development ()
    Kbrf      = input ('\n Enter form loss coefficient for a full fuel bundle (Kbrf []): '↙
);
    Kgsf      = input ('\n Enter form loss coefficient for a full fuel spacer (Kgsf []): '↙
);
    Kbrp      = input ('\n Enter form loss coefficient for a partial fuel bundle (Kbrp [] ↙
): ');
    Kgsp      = input ('\n Enter form loss coefficient for a partial fuel spacer (Kgsp [] ↙
): ');
else,
    Slaminlet = 109.174754;
    Slambrf   = 98.808304;
    Slamgsf   = 66.241816;
    Slambrp   = 98.035538;
    Slamgsp   = 43.359972;
    f1        = 0.96;
    Kbrinlet  = -4.268716;
    Kinlet    = (Kbrinlet + f1);
    Kbrf      = 0.300386;
    Kgsf      = 0.128565;
    Kbrp      = 0.930330;
    Kgsp      = 0.318634;
end;
disp    (' ');
disp    ('-------------------------------------------------------------------------');
disp    (' ');
disp    ('The stratification term for each component, based on CFD computational');
disp    ('experiments modeling buoyancy-induced flow in a simplified model      ');
disp    ('domain having similar geometric component length scales as a          ');
disp    ('prototypical 9x9 BWR fuel assembly, exist as defaults for this        ');
disp    ('program.  The defaults may not accurately represent these terms in an ');
disp    ('actual 9x9 BWR fuel assembly, but do provide reasonable starting      ');
disp    ('estimates when lacking specific measurement data.  Caution should be  ');
disp    ('exercised if using these defaults outside the net fuel assembly heat  ');
disp    ('input range of 300 to 1200 Watts or when using any input values above ');
disp    ('other than the default options; from which the stratification default ');
disp    ('values here are based.  Use these default values?                     ');
```

```
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt5    = input ('\n Enter Selection: ');
if (opt5 == 2),
    fprintf ('\n');
    Sbr1 = input ('\n Enter stratification parameter for br1 heat flux distribution (Sbr1 ↙
 []): ');
    Sgs1 = input ('\n Enter stratification parameter for gs1 heat flux distribution (Sgs1 ↙
 []): ');
    Sbr2 = input ('\n Enter stratification parameter for br2 heat flux distribution (Sbr2 ↙
 []): ');
    Sgs2 = input ('\n Enter stratification parameter for gs2 heat flux distribution (Sgs2 ↙
 []): ');
    Sbr3 = input ('\n Enter stratification parameter for br3 heat flux distribution (Sbr3 ↙
 []): ');
    Sgs3 = input ('\n Enter stratification parameter for gs3 heat flux distribution (Sgs3 ↙
 []): ');
    Sbr4 = input ('\n Enter stratification parameter for br4 heat flux distribution (Sbr4 ↙
 []): ');
    Sgs4 = input ('\n Enter stratification parameter for gs4 heat flux distribution (Sgs4 ↙
 []): ');
    Sbr5 = input ('\n Enter stratification parameter for br5 heat flux distribution (Sbr5 ↙
 []): ');
    Sgs5 = input ('\n Enter stratification parameter for gs5 heat flux distribution (Sgs5 ↙
 []): ');
    Sbr6 = input ('\n Enter stratification parameter for br6 heat flux distribution (Sbr6 ↙
 []): ');
    Sgs6 = input ('\n Enter stratification parameter for gs6 heat flux distribution (Sgs6 ↙
 []): ');
    Sbr7 = input ('\n Enter stratification parameter for br7 heat flux distribution (Sbr7 ↙
 []): ');
    Sgs7 = input ('\n Enter stratification parameter for gs7 heat flux distribution (Sgs7 ↙
 []): ');
    Sbr8 = input ('\n Enter stratification parameter for br8 heat flux distribution (Sbr8 ↙
 []): ');
else,
    Sbr1 = (0.503587922 + (-0.00028164*Qt))/(1 + (-0.00053886*Qt));
    Sbr2 = 0.49716088 + (-0.000000011934*(Qt^2)) + (0.724485688/Qt);
    Sbr3 = 0.499488751 + (-0.000018593*Qt) + 0.000279154*(Qt^(0.5));
    Sbr4 = 0.50185765 + (-0.000009626*Qt) + (-0.00000000000012093)*(Qt^3);
    Sbr5 = 0.512396949 + (-0.0000000030197)*(Qt^2) + (-0.00227177)*(log(Qt));
    Sbr6 = 0.509728768 + (-0.0000000000019446)*(Qt^3) + (-0.00207648)*(log(Qt));
    Sbr7 = 0.456567801 + (-0.0000000089221*(Qt^2)) + (0.0000000000000126145)*(Qt^4);
    Sbr8 = 0.458573563 + (-0.000000008727*(Qt^2)) + (0.0000000000000148783)*(Qt^4);
    Sgs1 = 0.495075667 + (-0.0000000020631*(Qt^2)) + (7.82398e-16)*(Qt^4);
    Sgs2 = 0.494977301 + (-0.00000000017207)*(Qt^(2.5)) + (0.0000000000047344)*(Qt^3);
    Sgs3 = 0.495179425 + (-0.00000021383)*(Qt)*(log(Qt)) + (0.000000000000947431)*(Qt^3);
    Sgs4 = 0.495337812 + (-0.0000021272)*(Qt) + (0.00000000169065)*(Qt^2);
    Sgs5 = 0.501793174 + (0.0000000611891)*(Qt^(1.5)) + (-0.00082825)*(log(Qt));
    Sgs6 = 0.531395932 + (0.00184623)*(exp(-Qt/-251.031455));
    Sgs7 = 0.535685871 + (0.001790102)*(exp(-Qt/-244.064916));
end;
disp    (' ');
disp    ('------------------------------------------------------------------------');
disp    (' ');
disp    ('Initial estimates of the mean air temperature in each assembly        ');
disp    ('component and the steady-state mass flow rate through the modeled     ');
disp    ('domain are required inputs, as well as several criterion values for   ');
disp    ('the solution iteration process.  If default options (or reasonably    ');
disp    ('similar values) are chosen for the prior inputs above and the net fuel');
```

```
disp    ('assembly heat input rate is within (or near) the range of 300 to 1200 ');
disp    ('Watts, the existing default values stored in this program for these   ');
disp    ('input parameters are sufficient for obtaining an accurate solution    ');
disp    ('at any net fuel assembly heat input rate considered.  Use these       ');
disp    ('default values?                                                       ');
disp    (' ');
disp    ('Options: 1 = Yes (default), 2 = No                                    ');
opt6    = input ('\n Enter Selection: ');
if (opt6 == 2),
    fprintf ('\n');
    Tbr1m  = input ('\n Enter initial estimate of br1 air mean temperature (Tbr1m [K]): ' ↙
);
    Tgs1m  = input ('\n Enter initial estimate of gs1 air mean temperature (Tgs1m [K]): ' ↙
);
    Tbr2m  = input ('\n Enter initial estimate of br2 air mean temperature (Tbr2m [K]): ' ↙
);
    Tgs2m  = input ('\n Enter initial estimate of gs2 air mean temperature (Tgs2m [K]): ' ↙
);
    Tbr3m  = input ('\n Enter initial estimate of br3 air mean temperature (Tbr3m [K]): ' ↙
);
    Tgs3m  = input ('\n Enter initial estimate of gs3 air mean temperature (Tgs3m [K]): ' ↙
);
    Tbr4m  = input ('\n Enter initial estimate of br4 air mean temperature (Tbr4m [K]): ' ↙
);
    Tgs4m  = input ('\n Enter initial estimate of gs4 air mean temperature (Tgs4m [K]): ' ↙
);
    Tbr5m  = input ('\n Enter initial estimate of br5 air mean temperature (Tbr5m [K]): ' ↙
);
    Tgs5m  = input ('\n Enter initial estimate of gs5 air mean temperature (Tgs5m [K]): ' ↙
);
    Tbr6m  = input ('\n Enter initial estimate of br6 air mean temperature (Tbr6m [K]): ' ↙
);
    Tgs6m  = input ('\n Enter initial estimate of gs6 air mean temperature (Tgs6m [K]): ' ↙
);
    Tbr7m  = input ('\n Enter initial estimate of br7 air mean temperature (Tbr7m [K]): ' ↙
);
    Tgs7m  = input ('\n Enter initial estimate of gs7 air mean temperature (Tgs7m [K]): ' ↙
);
    Tbr8m  = input ('\n Enter initial estimate of br8 air mean temperature (Tbr8m [K]): ' ↙
);
    Mdot_initial      = input ('\n Enter initial estimate of air mass flow rate (Mdot_ini ↙
tial [kg/s]): ');
    Tm_tolerance      = input ('\n Enter tolerance for convergence of mean temperatures ( ↙
Tm_tolerance [K]): ');
    Tm_max_its        = input ('\n Enter max number of iterations for estimating mean tem ↙
peratures (Tm_max_its []): ');
    Prop_Tm_tolerance = input ('\n Enter temperature tolerance for mean air properties (P ↙
rop_Tm_tolerance [K]): ');
    Prop_max_its      = input ('\n Enter max number of updates to air properties at new m ↙
ean temperatures (Prop_max_its []): ');
    fur_br = input ('\n Enter under-relaxation factor for updating fuel bundle mean tempe ↙
ratures (fur_br []): ');
    fur_gs = input ('\n Enter under-relaxation factor for updating fuel spacer mean tempe ↙
ratures (fur_gs []): ');
else,
    Tbr1m  = 330;
    Tgs1m  = 377;
    Tbr2m  = 418;
    Tgs2m  = 461;
    Tbr3m  = 501;
```

```
    Tgs3m  = 542;
    Tbr4m  = 582;
    Tgs4m  = 622;
    Tbr5m  = 661;
    Tgs5m  = 700;
    Tbr6m  = 735;
    Tgs6m  = 772;
    Tbr7m  = 803;
    Tgs7m  = 840;
    Tbr8m  = 870;
    Mdot_initial      = 0.00129512;
    Tm_tolerance      = 0.2;
    Tm_max_its        = 100;
    Prop_Tm_tolerance = 0.4;
    Prop_max_its      = 100;
    fur_br = 0.5;
    fur_gs = 0.5;
end;
fprintf ('\n');

%Call M-file function propertybuoyancy.m to calculate mean air properties at
%initial mean temperature estimates entered above.

Tmprop   = Tbr1m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br1m = rho_m;
Cp_br1m  = Cp_m;
mu_br1m  = mu_m;
nu_br1m  = nu_m;

Tmprop   = Tgs1m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs1m = rho_m;
Cp_gs1m  = Cp_m;
mu_gs1m  = mu_m;
nu_gs1m  = nu_m;

Tmprop   = Tbr2m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br2m = rho_m;
Cp_br2m  = Cp_m;
mu_br2m  = mu_m;
nu_br2m  = nu_m;

Tmprop   = Tgs2m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs2m = rho_m;
Cp_gs2m  = Cp_m;
mu_gs2m  = mu_m;
nu_gs2m  = nu_m;

Tmprop   = Tbr3m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br3m = rho_m;
Cp_br3m  = Cp_m;
mu_br3m  = mu_m;
nu_br3m  = nu_m;

Tmprop   = Tgs3m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
```

```
rho_gs3m = rho_m;
Cp_gs3m  = Cp_m;
mu_gs3m  = mu_m;
nu_gs3m  = nu_m;

Tmprop   = Tbr4m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br4m = rho_m;
Cp_br4m  = Cp_m;
mu_br4m  = mu_m;
nu_br4m  = nu_m;

Tmprop   = Tgs4m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs4m = rho_m;
Cp_gs4m  = Cp_m;
mu_gs4m  = mu_m;
nu_gs4m  = nu_m;

Tmprop   = Tbr5m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br5m = rho_m;
Cp_br5m  = Cp_m;
mu_br5m  = mu_m;
nu_br5m  = nu_m;

Tmprop   = Tgs5m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs5m = rho_m;
Cp_gs5m  = Cp_m;
mu_gs5m  = mu_m;
nu_gs5m  = nu_m;

Tmprop   = Tbr6m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br6m = rho_m;
Cp_br6m  = Cp_m;
mu_br6m  = mu_m;
nu_br6m  = nu_m;

Tmprop   = Tgs6m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs6m = rho_m;
Cp_gs6m  = Cp_m;
mu_gs6m  = mu_m;
nu_gs6m  = nu_m;

Tmprop   = Tbr7m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br7m = rho_m;
Cp_br7m  = Cp_m;
mu_br7m  = mu_m;
nu_br7m  = nu_m;

Tmprop   = Tgs7m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs7m = rho_m;
Cp_gs7m  = Cp_m;
mu_gs7m  = mu_m;
nu_gs7m  = nu_m;
```

```
Tmprop   = Tbr8m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br8m = rho_m;
Cp_br8m  = Cp_m;
mu_br8m  = mu_m;
nu_br8m  = nu_m;

%Verify that several design-specific parameters and critical input values are entered
%and stored correctly.

disp    ('------------------------------------------------------------------------');
disp    (' ');
disp    ('The following design-specific and critical input values are stored:   ');
fprintf ('\n');
fprintf ('Qt (W)              = %g\n', Qt);
fprintf ('Slaminlet ()        = %g\n', Slaminlet);
fprintf ('Slambrf ()          = %g\n', Slambrf);
fprintf ('Slamgsf ()          = %g\n', Slamgsf);
fprintf ('Slambrp ()          = %g\n', Slambrp);
fprintf ('Slamgsp ()          = %g\n', Slamgsp);
fprintf ('fl ()               = %g\n', fl);
fprintf ('Kinlet ()           = %g\n', Kinlet);
fprintf ('Kbrf ()             = %g\n', Kbrf);
fprintf ('Kgsf ()             = %g\n', Kgsf);
fprintf ('Kbrp ()             = %g\n', Kbrp);
fprintf ('Kgsp ()             = %g\n', Kgsp);
fprintf ('Mdot_initial (kg/s) = %g\n', Mdot_initial);
fprintf ('\n');

%%%INITIALIZE ITS COUNT WITH STARTING VALUES AND BEGIN OUTER LOOP FOR UPDATING
%%%AIR PROPERTIES AT CONVERGED (Tm) VALUES FROM INNER LOOP%%%

Prop_its_count       = 1;
Prop_Tbr1m_update(1) = Tbr1m;  %Units are (K)
Prop_Tgs1m_update(1) = Tgs1m;  %Units are (K)
Prop_Tbr2m_update(1) = Tbr2m;  %Units are (K)
Prop_Tgs2m_update(1) = Tgs2m;  %Units are (K)
Prop_Tbr3m_update(1) = Tbr3m;  %Units are (K)
Prop_Tgs3m_update(1) = Tgs3m;  %Units are (K)
Prop_Tbr4m_update(1) = Tbr4m;  %Units are (K)
Prop_Tgs4m_update(1) = Tgs4m;  %Units are (K)
Prop_Tbr5m_update(1) = Tbr5m;  %Units are (K)
Prop_Tgs5m_update(1) = Tgs5m;  %Units are (K)
Prop_Tbr6m_update(1) = Tbr6m;  %Units are (K)
Prop_Tgs6m_update(1) = Tgs6m;  %Units are (K)
Prop_Tbr7m_update(1) = Tbr7m;  %Units are (K)
Prop_Tgs7m_update(1) = Tgs7m;  %Units are (K)
Prop_Tbr8m_update(1) = Tbr8m;  %Units are (K)
Prop_Mdot_update(1)  = Mdot_initial;  %Units are (kg/s)

for z = 2:Prop_max_its,

    %%%INITIALIZE ITS COUNT AND START INNER LOOP FOR APPROXIMATING (Tm) SOLUTIONS%%%

    Tm_its_count = 1;

    for j = 2:Tm_max_its,
        fprintf ('\n');
        disp    ('------------------------------------------------------------------');
```

```
        fprintf ('\n');
        fprintf ('Starting value of (Mdot) for inner loop (kg/s) = %g\n', Mdot_initial);

        %%%START SUB-CALCULATION FOR (Mdot) SOLUTION(S) AT CURRENT (Tm) VALUES%%%

        %Call M-file function massflowbuoyancy.m to approximate (Mdot) zero(s) of
        %the governing function, display (Mdot) root results, and the final (Mdot)
        %value assumed.

        [Mdot_final] = massflowbuoyancy;

        %Display current and prior (Mdot) solution estimates for comparison and to
        %also check the progress of the solution procedure.

        fprintf ('\n');
        fprintf ('Previous inner loop estimate of (Mdot) value (kg/s) = %g\n', Mdot_initi ↙
al);
        fprintf ('Current inner loop estimate of (Mdot) value (kg/s)  = %g\n\n', Mdot_fin ↙
al);

        %%%END SUB-CALCULATION FOR (Mdot) SOLUTION(S) AT CURRENT (Tm) VALUES%%%

        %Calculate and display current (Tm) inner loop approximation results, check
        %for convergence, and (if needed) update to better estimates of (Tm) values.
        %Use current (Mdot) loop solution as the new (Mdot) initial loop value.

        Tbr8m_energy_balance = ((Qbrp*Sbr8)/Cp_br8m + Qgsp/Cp_gs7m + Qbrp/Cp_br7m +...
            Qgsp/Cp_gs6m + Qbrp/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m...
            + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m...
            + Qgsf/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tgs7m_energy_balance = ((Qgsp*Sgs7)/Cp_gs7m + Qbrp/Cp_br7m + Qgsp/Cp_gs6m +...
            Qbrp/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m...
            + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m...
            + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tbr7m_energy_balance = ((Qbrp*Sbr7)/Cp_br7m + Qgsp/Cp_gs6m + Qbrp/Cp_br6m +...
            Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m...
            + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m +...
            Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tgs6m_energy_balance = ((Qgsp*Sgs6)/Cp_gs6m + Qbrp/Cp_br6m + Qgsf/Cp_gs5m +...
            Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m...
            + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + ...
            Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tbr6m_energy_balance = ((Qbrp*Sbr6)/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m +...
            Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m...
            + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) ...
            + Tin;  %Units are (K)
        Tgs5m_energy_balance = ((Qgsf*Sgs5)/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m +...
            Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m...
            + Qgsf/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tbr5m_energy_balance = ((Qbrf*Sbr5)/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m +...
            Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m...
            + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tgs4m_energy_balance = ((Qgsf*Sgs4)/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m +...
            Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m...
            + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tbr4m_energy_balance = ((Qbrf*Sbr4)/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m +...
            Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + ...
            Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
        Tgs3m_energy_balance = ((Qgsf*Sgs3)/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m +...
            Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) ...
```

```
        + Tin;  %Units are (K)
    Tbr3m_energy_balance = ((Qbrf*Sbr3)/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m +...
        Qgsf/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
    Tgs2m_energy_balance = ((Qgsf*Sgs2)/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m +...
        Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
    Tbr2m_energy_balance = ((Qbrf*Sbr2)/Cp_br2m + Qgsf/Cp_gs1m +...
        Qbrf/Cp_br1m)*(1/Mdot_final) + Tin;  %Units are (K)
    Tgs1m_energy_balance = ((Qgsf*Sgs1)/Cp_gs1m + Qbrf/Cp_br1m)*(1/Mdot_final) ...
        + Tin;  %Units are (K)
    Tbr1m_energy_balance = ((Qbrf*Sbr1)/Cp_br1m)*(1/Mdot_final) ...
        + Tin;  %Units are (K)
    fprintf ('(Tbr1m) value from current energy balance (K)    = %g\n', Tbr1m_energy_ ↙
balance);
    fprintf ('(Tbr1m) value from previous iteration update (K) = %g\n', Tbr1m);
    fprintf ('(Tbr8m) value from current energy balance (K)    = %g\n', Tbr8m_energy_ ↙
balance);
    fprintf ('(Tbr8m) value from previous iteration update (K) = %g\n\n', Tbr8m);
    if (abs(Tbr8m_energy_balance - Tbr8m) < Tm_tolerance ...
            & abs(Tgs7m_energy_balance - Tgs7m) < Tm_tolerance ...
            & abs(Tbr7m_energy_balance - Tbr7m) < Tm_tolerance ...
            & abs(Tgs6m_energy_balance - Tgs6m) < Tm_tolerance ...
            & abs(Tbr6m_energy_balance - Tbr6m) < Tm_tolerance ...
            & abs(Tgs5m_energy_balance - Tgs5m) < Tm_tolerance ...
            & abs(Tbr5m_energy_balance - Tbr5m) < Tm_tolerance ...
            & abs(Tgs4m_energy_balance - Tgs4m) < Tm_tolerance ...
            & abs(Tbr4m_energy_balance - Tbr4m) < Tm_tolerance ...
            & abs(Tgs3m_energy_balance - Tgs3m) < Tm_tolerance ...
            & abs(Tbr3m_energy_balance - Tbr3m) < Tm_tolerance ...
            & abs(Tgs2m_energy_balance - Tgs2m) < Tm_tolerance ...
            & abs(Tbr2m_energy_balance - Tbr2m) < Tm_tolerance ...
            & abs(Tgs1m_energy_balance - Tgs1m) < Tm_tolerance ...
            & abs(Tbr1m_energy_balance - Tbr1m) < Tm_tolerance),
        disp (' ');
        disp ('------------------------------------------------------------------');
        disp (' ');
        disp ('Inner loop (Tm) iterations have successfully converged!          ');
        disp (' ');
        break;
    end;
    Tbr8m_update(j) = Tbr8m + (fur_br)*(Tbr8m_energy_balance - Tbr8m);  %Units are (K)
    Tgs7m_update(j) = Tgs7m + (fur_gs)*(Tgs7m_energy_balance - Tgs7m);  %Units are (K)
    Tbr7m_update(j) = Tbr7m + (fur_br)*(Tbr7m_energy_balance - Tbr7m);  %Units are (K)
    Tgs6m_update(j) = Tgs6m + (fur_gs)*(Tgs6m_energy_balance - Tgs6m);  %Units are (K)
    Tbr6m_update(j) = Tbr6m + (fur_br)*(Tbr6m_energy_balance - Tbr6m);  %Units are (K)
    Tgs5m_update(j) = Tgs5m + (fur_gs)*(Tgs5m_energy_balance - Tgs5m);  %Units are (K)
    Tbr5m_update(j) = Tbr5m + (fur_br)*(Tbr5m_energy_balance - Tbr5m);  %Units are (K)
    Tgs4m_update(j) = Tgs4m + (fur_gs)*(Tgs4m_energy_balance - Tgs4m);  %Units are (K)
    Tbr4m_update(j) = Tbr4m + (fur_br)*(Tbr4m_energy_balance - Tbr4m);  %Units are (K)
    Tgs3m_update(j) = Tgs3m + (fur_gs)*(Tgs3m_energy_balance - Tgs3m);  %Units are (K)
    Tbr3m_update(j) = Tbr3m + (fur_br)*(Tbr3m_energy_balance - Tbr3m);  %Units are (K)
    Tgs2m_update(j) = Tgs2m + (fur_gs)*(Tgs2m_energy_balance - Tgs2m);  %Units are (K)
    Tbr2m_update(j) = Tbr2m + (fur_br)*(Tbr2m_energy_balance - Tbr2m);  %Units are (K)
    Tgs1m_update(j) = Tgs1m + (fur_gs)*(Tgs1m_energy_balance - Tgs1m);  %Units are (K)
    Tbr1m_update(j) = Tbr1m + (fur_br)*(Tbr1m_energy_balance - Tbr1m);  %Units are (K)
    Tbr8m           = Tbr8m_update(j);
    Tgs7m           = Tgs7m_update(j);
    Tbr7m           = Tbr7m_update(j);
    Tgs6m           = Tgs6m_update(j);
    Tbr6m           = Tbr6m_update(j);
    Tgs5m           = Tgs5m_update(j);
```

```
    Tbr5m           = Tbr5m_update(j);
    Tgs4m           = Tgs4m_update(j);
    Tbr4m           = Tbr4m_update(j);
    Tgs3m           = Tgs3m_update(j);
    Tbr3m           = Tbr3m_update(j);
    Tgs2m           = Tgs2m_update(j);
    Tbr2m           = Tbr2m_update(j);
    Tgs1m           = Tgs1m_update(j);
    Tbr1m           = Tbr1m_update(j);
    Mdot_update(j)  = Mdot_final;  %Units are (kg/s)
    Mdot_initial    = Mdot_final;
    Tm_its_count    = j;
end;

%%%END INNER LOOP FOR APPROXIMATING (Tm) SOLUTIONS%%%

%Check for non-convergence of inner (Tm) loop estimations.  If inner
%(Tm) loop fails to converge, display results for the last several loop
%iterations for one or two select parameters.

if (Tm_its_count >= Tm_max_its),
    disp (' ');
    disp ('-----------------------------------------------------------------------');
    disp (' ');
    disp ('Inner loop (Tm) solutions not found within tolerance requirements!!');
    disp (' ');
    disp ('Selected parameter results over last few iterations are as follows:');
    disp (' ');
    fprintf ('(Tbr8m) value for (Mdot) approximation (K) = %g\n', Tbr8m_update(j-4));
    fprintf ('(Tbr8m) value for (Mdot) approximation (K) = %g\n', Tbr8m_update(j-3));
    fprintf ('(Tbr8m) value for (Mdot) approximation (K) = %g\n', Tbr8m_update(j-2));
    fprintf ('(Tbr8m) value for (Mdot) approximation (K) = %g\n', Tbr8m_update(j-1));
    fprintf ('(Tbr8m) value for (Mdot) approximation (K) = %g\n\n', Tbr8m_update(j));
    fprintf ('(Mdot) inner loop approximation (kg/s) = %g\n', Mdot_update(j-4));
    fprintf ('(Mdot) inner loop approximation (kg/s) = %g\n', Mdot_update(j-3));
    fprintf ('(Mdot) inner loop approximation (kg/s) = %g\n', Mdot_update(j-2));
    fprintf ('(Mdot) inner loop approximation (kg/s) = %g\n', Mdot_update(j-1));
    fprintf ('(Mdot) inner loop approximation (kg/s) = %g\n\n', Mdot_update(j));
    disp (' ');
end;

%Display final results from inner (Tm) loop estimates for (Mdot) as well as
%the air mean temperature and velocity in several fuel assembly components for
%comparison / monitoring of the solution progress.

Ubr1m = Mdot_final/(rho_br1m*Acbrf);  %Units are (m/s)
Ubr8m = Mdot_final/(rho_br8m*Acbrp);  %Units are (m/s)
fprintf ('Final inner loop br1 air mean temperature (K) = %g\n', Tbr1m);
fprintf ('Final inner loop br8 air mean temperature (K) = %g\n', Tbr8m);
fprintf ('Final inner loop br1 mean air velocity (m/s)  = %g\n', Ubr1m);
fprintf ('Final inner loop br8 mean air velocity (m/s)  = %g\n', Ubr8m);
fprintf ('Final inner loop air mass flow rate (kg/s)    = %g\n\n', Mdot_final);

%Check outer air properties update loop for convergence, then (if
%needed) update to a better estimate of the air properties at final (Tm)
%solutions from inner loop.  Also, use last (Tm) inner loop solutions as
%the new inner loop initial (Tm) values including (Mdot).

Prop_Tbr1m_update(z) = Tbr1m;  %Units are (K)
Prop_Tgs1m_update(z) = Tgs1m;  %Units are (K)
```

```
    Prop_Tbr2m_update(z) = Tbr2m;  %Units are (K)
    Prop_Tgs2m_update(z) = Tgs2m;  %Units are (K)
    Prop_Tbr3m_update(z) = Tbr3m;  %Units are (K)
    Prop_Tgs3m_update(z) = Tgs3m;  %Units are (K)
    Prop_Tbr4m_update(z) = Tbr4m;  %Units are (K)
    Prop_Tgs4m_update(z) = Tgs4m;  %Units are (K)
    Prop_Tbr5m_update(z) = Tbr5m;  %Units are (K)
    Prop_Tgs5m_update(z) = Tgs5m;  %Units are (K)
    Prop_Tbr6m_update(z) = Tbr6m;  %Units are (K)
    Prop_Tgs6m_update(z) = Tgs6m;  %Units are (K)
    Prop_Tbr7m_update(z) = Tbr7m;  %Units are (K)
    Prop_Tgs7m_update(z) = Tgs7m;  %Units are (K)
    Prop_Tbr8m_update(z) = Tbr8m;  %Units are (K)
    Prop_Mdot_update(z)  = Mdot_final;  %Units are (kg/s)
    Prop_Ubr1m_update(z) = Ubr1m;  %Units are (m/s)
    Prop_Ubr8m_update(z) = Ubr8m;  %Units are (m/s)
    if (abs(Prop_Tbr8m_update(z) - Prop_Tbr8m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs7m_update(z) - Prop_Tgs7m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr7m_update(z) - Prop_Tbr7m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs6m_update(z) - Prop_Tgs6m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr6m_update(z) - Prop_Tbr6m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs5m_update(z) - Prop_Tgs5m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr5m_update(z) - Prop_Tbr5m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs4m_update(z) - Prop_Tgs4m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr4m_update(z) - Prop_Tbr4m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs3m_update(z) - Prop_Tgs3m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr3m_update(z) - Prop_Tbr3m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs2m_update(z) - Prop_Tgs2m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr2m_update(z) - Prop_Tbr2m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tgs1m_update(z) - Prop_Tgs1m_update(z-1)) < Prop_Tm_tolerance ...
            & abs(Prop_Tbr1m_update(z) - Prop_Tbr1m_update(z-1)) < Prop_Tm_tolerance),
        disp (' ');
        disp ('------------------------------------------------------------------');
        disp (' ');
        disp ('Outer air properties update loop has converged at (Tm) values!  ');
        break;
    end;

    %Display previous outer property loop approximation for (Mdot) as well as
    %the current outer property loop approximation for (Mdot) for comparison /
    %solution progress monitoring.

    disp (' ');
    disp ('------------------------------------------------------------------');
    disp (' ');
    fprintf ('Previous outer property loop (Mdot) value (kg/s) = %g\n', Prop_Mdot_update( ↙
z-1));
    fprintf ('Current outer property loop (Mdot) value (kg/s)  = %g\n\n', Prop_Mdot_updat ↙
e(z));

    %Call M-file function propertybuoyancy.m to calculate mean air properties at
    %new mean temperature estimates from converged (Tm) inner loop solutions.

    Tmprop   = Tbr1m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_br1m = rho_m;
    Cp_br1m  = Cp_m;
    mu_br1m  = mu_m;
    nu_br1m  = nu_m;
```

```
Tmprop   = Tgs1m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs1m = rho_m;
Cp_gs1m  = Cp_m;
mu_gs1m  = mu_m;
nu_gs1m  = nu_m;

Tmprop   = Tbr2m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br2m = rho_m;
Cp_br2m  = Cp_m;
mu_br2m  = mu_m;
nu_br2m  = nu_m;

Tmprop   = Tgs2m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs2m = rho_m;
Cp_gs2m  = Cp_m;
mu_gs2m  = mu_m;
nu_gs2m  = nu_m;

Tmprop   = Tbr3m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br3m = rho_m;
Cp_br3m  = Cp_m;
mu_br3m  = mu_m;
nu_br3m  = nu_m;

Tmprop   = Tgs3m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs3m = rho_m;
Cp_gs3m  = Cp_m;
mu_gs3m  = mu_m;
nu_gs3m  = nu_m;

Tmprop   = Tbr4m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br4m = rho_m;
Cp_br4m  = Cp_m;
mu_br4m  = mu_m;
nu_br4m  = nu_m;

Tmprop   = Tgs4m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs4m = rho_m;
Cp_gs4m  = Cp_m;
mu_gs4m  = mu_m;
nu_gs4m  = nu_m;

Tmprop   = Tbr5m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_br5m = rho_m;
Cp_br5m  = Cp_m;
mu_br5m  = mu_m;
nu_br5m  = nu_m;

Tmprop   = Tgs5m;
[rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
rho_gs5m = rho_m;
Cp_gs5m  = Cp_m;
```

```
    mu_gs5m  = mu_m;
    nu_gs5m  = nu_m;

    Tmprop   = Tbr6m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_br6m = rho_m;
    Cp_br6m  = Cp_m;
    mu_br6m  = mu_m;
    nu_br6m  = nu_m;

    Tmprop   = Tgs6m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_gs6m = rho_m;
    Cp_gs6m  = Cp_m;
    mu_gs6m  = mu_m;
    nu_gs6m  = nu_m;

    Tmprop   = Tbr7m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_br7m = rho_m;
    Cp_br7m  = Cp_m;
    mu_br7m  = mu_m;
    nu_br7m  = nu_m;

    Tmprop   = Tgs7m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_gs7m = rho_m;
    Cp_gs7m  = Cp_m;
    mu_gs7m  = mu_m;
    nu_gs7m  = nu_m;

    Tmprop   = Tbr8m;
    [rho_m, Cp_m, mu_m, nu_m] = propertybuoyancy;
    rho_br8m = rho_m;
    Cp_br8m  = Cp_m;
    mu_br8m  = mu_m;
    nu_br8m  = nu_m;

    Mdot_initial   = Mdot_final;
    Prop_its_count = z;
end;

%%%END OUTER LOOP FOR UPDATING AIR PROPERTIES AT CONVERGED INNER LOOP (Tm) VALUES%%%

%Check for non-convergence of outer air properties update loop.  If outer
%loop fails to converge, display results for the last several outer loop
%iterations for one or two select parameters.

if (Prop_its_count >= Prop_max_its),
    disp (' ');
    disp ('-----------------------------------------------------------------');
    disp (' ');
    disp ('Outer air properties update loop at (Tm) values has not converged!!');
    disp (' ');
    disp ('Selected parameter results over last few iterations are as follows:');
    disp (' ');
    fprintf ('(Tbr8m) outer loop value approximation (K) = %g\n', Prop_Tbr8m_update(z-4));
    fprintf ('(Tbr8m) outer loop value approximation (K) = %g\n', Prop_Tbr8m_update(z-3));
    fprintf ('(Tbr8m) outer loop value approximation (K) = %g\n', Prop_Tbr8m_update(z-2));
    fprintf ('(Tbr8m) outer loop value approximation (K) = %g\n', Prop_Tbr8m_update(z-1));
```

```
    fprintf ('(Tbr8m) outer loop value approximation (K) = %g\n\n', Prop_Tbr8m_update(z));
    fprintf ('(Mdot) outer loop approximation (kg/s) = %g\n', Prop_Mdot_update(z-4));
    fprintf ('(Mdot) outer loop approximation (kg/s) = %g\n', Prop_Mdot_update(z-3));
    fprintf ('(Mdot) outer loop approximation (kg/s) = %g\n', Prop_Mdot_update(z-2));
    fprintf ('(Mdot) outer loop approximation (kg/s) = %g\n', Prop_Mdot_update(z-1));
    fprintf ('(Mdot) outer loop approximation (kg/s) = %g\n\n', Prop_Mdot_update(z));
    fprintf ('(Ubr1m) at update of air properties (m/s) = %g\n', Prop_Ubr1m_update(z-4));
    fprintf ('(Ubr1m) at update of air properties (m/s) = %g\n', Prop_Ubr1m_update(z-3));
    fprintf ('(Ubr1m) at update of air properties (m/s) = %g\n', Prop_Ubr1m_update(z-2));
    fprintf ('(Ubr1m) at update of air properties (m/s) = %g\n', Prop_Ubr1m_update(z-1));
    fprintf ('(Ubr1m) at update of air properties (m/s) = %g\n\n', Prop_Ubr1m_update(z));
    fprintf ('(Ubr8m) at update of air properties (m/s) = %g\n', Prop_Ubr8m_update(z-4));
    fprintf ('(Ubr8m) at update of air properties (m/s) = %g\n', Prop_Ubr8m_update(z-3));
    fprintf ('(Ubr8m) at update of air properties (m/s) = %g\n', Prop_Ubr8m_update(z-2));
    fprintf ('(Ubr8m) at update of air properties (m/s) = %g\n', Prop_Ubr8m_update(z-1));
    fprintf ('(Ubr8m) at update of air properties (m/s) = %g\n\n', Prop_Ubr8m_update(z));
    disp (' ');
end;

%Calculate and display final results from this model run for mean air
%temperatures (Tm), exit air temperatures (Tex), mean air velocities (Um), air
%mass flow rate (Mdot), and total pressure loss across the entire BWR fuel
%assembly (P_loss).

Tbr1ex = (Tbr1m - Tin)/Sbr1 + Tin;        %Units are (K)
Tgs1ex = (Tgs1m - Tbr1ex)/Sgs1 + Tbr1ex;  %Units are (K)
Tbr2ex = (Tbr2m - Tgs1ex)/Sbr2 + Tgs1ex;  %Units are (K)
Tgs2ex = (Tgs2m - Tbr2ex)/Sgs2 + Tbr2ex;  %Units are (K)
Tbr3ex = (Tbr3m - Tgs2ex)/Sbr3 + Tgs2ex;  %Units are (K)
Tgs3ex = (Tgs3m - Tbr3ex)/Sgs3 + Tbr3ex;  %Units are (K)
Tbr4ex = (Tbr4m - Tgs3ex)/Sbr4 + Tgs3ex;  %Units are (K)
Tgs4ex = (Tgs4m - Tbr4ex)/Sgs4 + Tbr4ex;  %Units are (K)
Tbr5ex = (Tbr5m - Tgs4ex)/Sbr5 + Tgs4ex;  %Units are (K)
Tgs5ex = (Tgs5m - Tbr5ex)/Sgs5 + Tbr5ex;  %Units are (K)
Tbr6ex = (Tbr6m - Tgs5ex)/Sbr6 + Tgs5ex;  %Units are (K)
Tgs6ex = (Tgs6m - Tbr6ex)/Sgs6 + Tbr6ex;  %Units are (K)
Tbr7ex = (Tbr7m - Tgs6ex)/Sbr7 + Tgs6ex;  %Units are (K)
Tgs7ex = (Tgs7m - Tbr7ex)/Sgs7 + Tbr7ex;  %Units are (K)
Tbr8ex = (Tbr8m - Tgs7ex)/Sbr8 + Tgs7ex;  %Units are (K)
Ubr1m  = Mdot_final/(rho_br1m*Acbrf);     %Units are (m/s)
Ubr2m  = Mdot_final/(rho_br2m*Acbrf);     %Units are (m/s)
Ubr3m  = Mdot_final/(rho_br3m*Acbrf);     %Units are (m/s)
Ubr4m  = Mdot_final/(rho_br4m*Acbrf);     %Units are (m/s)
Ubr5m  = Mdot_final/(rho_br5m*Acbrf);     %Units are (m/s)
Ubr6m  = Mdot_final/(rho_br6m*Acbrp);     %Units are (m/s)
Ubr7m  = Mdot_final/(rho_br7m*Acbrp);     %Units are (m/s)
Ubr8m  = Mdot_final/(rho_br8m*Acbrp);     %Units are (m/s)
Ugs1m  = Mdot_final/(rho_gs1m*Acgsf);     %Units are (m/s)
Ugs2m  = Mdot_final/(rho_gs2m*Acgsf);     %Units are (m/s)
Ugs3m  = Mdot_final/(rho_gs3m*Acgsf);     %Units are (m/s)
Ugs4m  = Mdot_final/(rho_gs4m*Acgsf);     %Units are (m/s)
Ugs5m  = Mdot_final/(rho_gs5m*Acgsf);     %Units are (m/s)
Ugs6m  = Mdot_final/(rho_gs6m*Acgsp);     %Units are (m/s)
Ugs7m  = Mdot_final/(rho_gs7m*Acgsp);     %Units are (m/s)
P_loss = (g*(Lbr*((rho_not - rho_br1m) + (rho_not - rho_br2m) + (rho_not - rho_br3m) ...
    + (rho_not - rho_br4m) + (rho_not - rho_br5m) + (rho_not - rho_br6m) ...
    + (rho_not - rho_br7m) + (rho_not - rho_br8m)) + Lgs*((rho_not - rho_gs1m) ...
    + (rho_not - rho_gs2m) + (rho_not - rho_gs3m) + (rho_not - rho_gs4m) ...
    + (rho_not - rho_gs5m) + (rho_not - rho_gs6m) ...
    + (rho_not - rho_gs7m))));  %Units are (Pa)
```

```
disp (' ');
disp ('-----------------------------------------------------------------');
disp (' ');
disp ('Final Model Results:                                              ');
disp (' ');
fprintf ('Final assembly air mass flow rate (kg/s) = %g\n', Mdot_final);
fprintf ('Final assembly exit air temperature (K)  = %g\n', Tbr8ex);
fprintf ('Final gs7 exit air temperature (K)       = %g\n', Tgs7ex);
fprintf ('Final br7 exit air temperature (K)       = %g\n', Tbr7ex);
fprintf ('Final gs6 exit air temperature (K)       = %g\n', Tgs6ex);
fprintf ('Final br6 exit air temperature (K)       = %g\n', Tbr6ex);
fprintf ('Final gs5 exit air temperature (K)       = %g\n', Tgs5ex);
fprintf ('Final br5 exit air temperature (K)       = %g\n', Tbr5ex);
fprintf ('Final gs4 exit air temperature (K)       = %g\n', Tgs4ex);
fprintf ('Final br4 exit air temperature (K)       = %g\n', Tbr4ex);
fprintf ('Final gs3 exit air temperature (K)       = %g\n', Tgs3ex);
fprintf ('Final br3 exit air temperature (K)       = %g\n', Tbr3ex);
fprintf ('Final gs2 exit air temperature (K)       = %g\n', Tgs2ex);
fprintf ('Final br2 exit air temperature (K)       = %g\n', Tbr2ex);
fprintf ('Final gs1 exit air temperature (K)       = %g\n', Tgs1ex);
fprintf ('Final br1 exit air temperature (K)       = %g\n', Tbr1ex);
fprintf ('Final br8 mean air temperature (K)       = %g\n', Tbr8m);
fprintf ('Final gs7 mean air temperature (K)       = %g\n', Tgs7m);
fprintf ('Final br7 mean air temperature (K)       = %g\n', Tbr7m);
fprintf ('Final gs6 mean air temperature (K)       = %g\n', Tgs6m);
fprintf ('Final br6 mean air temperature (K)       = %g\n', Tbr6m);
fprintf ('Final gs5 mean air temperature (K)       = %g\n', Tgs5m);
fprintf ('Final br5 mean air temperature (K)       = %g\n', Tbr5m);
fprintf ('Final gs4 mean air temperature (K)       = %g\n', Tgs4m);
fprintf ('Final br4 mean air temperature (K)       = %g\n', Tbr4m);
fprintf ('Final gs3 mean air temperature (K)       = %g\n', Tgs3m);
fprintf ('Final br3 mean air temperature (K)       = %g\n', Tbr3m);
fprintf ('Final gs2 mean air temperature (K)       = %g\n', Tgs2m);
fprintf ('Final br2 mean air temperature (K)       = %g\n', Tbr2m);
fprintf ('Final gs1 mean air temperature (K)       = %g\n', Tgs1m);
fprintf ('Final br1 mean air temperature (K)       = %g\n', Tbr1m);
fprintf ('Final br8 mean air velocity (m/s)        = %g\n', Ubr8m);
fprintf ('Final gs7 mean air velocity (m/s)        = %g\n', Ugs7m);
fprintf ('Final br7 mean air velocity (m/s)        = %g\n', Ubr7m);
fprintf ('Final gs6 mean air velocity (m/s)        = %g\n', Ugs6m);
fprintf ('Final br6 mean air velocity (m/s)        = %g\n', Ubr6m);
fprintf ('Final gs5 mean air velocity (m/s)        = %g\n', Ugs5m);
fprintf ('Final br5 mean air velocity (m/s)        = %g\n', Ubr5m);
fprintf ('Final gs4 mean air velocity (m/s)        = %g\n', Ugs4m);
fprintf ('Final br4 mean air velocity (m/s)        = %g\n', Ubr4m);
fprintf ('Final gs3 mean air velocity (m/s)        = %g\n', Ugs3m);
fprintf ('Final br3 mean air velocity (m/s)        = %g\n', Ubr3m);
fprintf ('Final gs2 mean air velocity (m/s)        = %g\n', Ugs2m);
fprintf ('Final br2 mean air velocity (m/s)        = %g\n', Ubr2m);
fprintf ('Final gs1 mean air velocity (m/s)        = %g\n', Ugs1m);
fprintf ('Final br1 mean air velocity (m/s)        = %g\n\n', Ubr1m);
fprintf ('Final total pressure loss across assembly (Pa) = %g\n\n', P_loss);

disp (' ');
disp ('-----------------------------------------------------------------');
disp ('End Model Run                                                     ');
disp ('-----------------------------------------------------------------');
disp (' ');
fprintf ('\n\n');
```

< MASSFLOWBUOYANCY.M >

```
function [Mdot_root] = massflowbuoyancy

%M-file function massflowbuoyancy.m that estimates (Mdot) zero(s) of the
%governing function with user inputs and criteria from M-file script
%buoyancy.m, which then returns a final (Mdot) result value.

%Declare static global variables needed within this function or sub-functions
%called by this function.

global Sbr1;
global Sgs1;
global Sbr2;
global Sgs2;
global Sbr3;
global Sgs3;
global Sbr4;
global Sgs4;
global Sbr5;
global Sgs5;
global Sbr6;
global Sgs6;
global Sbr7;
global Sgs7;
global Sbr8;
global g;
global Dbrf;
global Dbrp;
global Dgsf;
global Dgsp;
global Lbr;
global Lgs;
global Acbrf;
global Acbrp;
global Acgsf;
global Acgsp;
global Slaminlet;
global Slambrf;
global Slamgsf;
global Slambrp;
global Slamgsp;
global Kinlet;
global Kbrf;
global Kgsf;
global Kbrp;
global Kgsp;
global Tin;
global Patm;
global Rgas;
global Qbrf;
global Qgsf;
global Qbrp;
global Qgsp;
global rho_not;

%Declare varying global variables needed within this function or sub-functions
%called by this function.

global Tbr1m;
global Tgs1m;
global Tbr2m;
```

```
global Tgs2m;
global Tbr3m;
global Tgs3m;
global Tbr4m;
global Tgs4m;
global Tbr5m;
global Tgs5m;
global Tbr6m;
global Tgs6m;
global Tbr7m;
global Tgs7m;
global Tbr8m;
global rho_br1m;
global Cp_br1m;
global mu_br1m;
global nu_br1m;
global rho_gs1m;
global Cp_gs1m;
global mu_gs1m;
global nu_gs1m;
global rho_br2m;
global Cp_br2m;
global mu_br2m;
global nu_br2m;
global rho_gs2m;
global Cp_gs2m;
global mu_gs2m;
global nu_gs2m;
global rho_br3m;
global Cp_br3m;
global mu_br3m;
global nu_br3m;
global rho_gs3m;
global Cp_gs3m;
global mu_gs3m;
global nu_gs3m;
global rho_br4m;
global Cp_br4m;
global mu_br4m;
global nu_br4m;
global rho_gs4m;
global Cp_gs4m;
global mu_gs4m;
global nu_gs4m;
global rho_br5m;
global Cp_br5m;
global mu_br5m;
global nu_br5m;
global rho_gs5m;
global Cp_gs5m;
global mu_gs5m;
global nu_gs5m;
global rho_br6m;
global Cp_br6m;
global mu_br6m;
global nu_br6m;
global rho_gs6m;
global Cp_gs6m;
global mu_gs6m;
global nu_gs6m;
```

```
global rho_br7m;
global Cp_br7m;
global mu_br7m;
global nu_br7m;
global rho_gs7m;
global Cp_gs7m;
global mu_gs7m;
global nu_gs7m;
global rho_br8m;
global Cp_br8m;
global mu_br8m;
global nu_br8m;
global Mdot_initial;

%Calculate any intermediate grouped parameter values necessary for defining
%final polynomial coefficients used by the 'roots(p)' function.

Tbr8delta = ((Qbrp*Sbr8)/Cp_br8m + Qgsp/Cp_gs7m + Qbrp/Cp_br7m + Qgsp/Cp_gs6m ...
    + Qbrp/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m ...
    + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m ...
    + Qbrf/Cp_br1m);
Tgs7delta = ((Qgsp*Sgs7)/Cp_gs7m + Qbrp/Cp_br7m + Qgsp/Cp_gs6m + Qbrp/Cp_br6m ...
    + Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m ...
    + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tbr7delta = ((Qbrp*Sbr7)/Cp_br7m + Qgsp/Cp_gs6m + Qbrp/Cp_br6m + Qgsf/Cp_gs5m ...
    + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m ...
    + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tgs6delta = ((Qgsp*Sgs6)/Cp_gs6m + Qbrp/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m ...
    + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m ...
    + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tbr6delta = ((Qbrp*Sbr6)/Cp_br6m + Qgsf/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m ...
    + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m ...
    + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tgs5delta = ((Qgsf*Sgs5)/Cp_gs5m + Qbrf/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m ...
    + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m ...
    + Qbrf/Cp_br1m);
Tbr5delta = ((Qbrf*Sbr5)/Cp_br5m + Qgsf/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m ...
    + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tgs4delta = ((Qgsf*Sgs4)/Cp_gs4m + Qbrf/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m ...
    + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tbr4delta = ((Qbrf*Sbr4)/Cp_br4m + Qgsf/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m ...
    + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tgs3delta = ((Qgsf*Sgs3)/Cp_gs3m + Qbrf/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m ...
    + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tbr3delta = ((Qbrf*Sbr3)/Cp_br3m + Qgsf/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m ...
    + Qbrf/Cp_br1m);
Tgs2delta = ((Qgsf*Sgs2)/Cp_gs2m + Qbrf/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tbr2delta = ((Qbrf*Sbr2)/Cp_br2m + Qgsf/Cp_gs1m + Qbrf/Cp_br1m);
Tgs1delta = ((Qgsf*Sgs1)/Cp_gs1m + Qbrf/Cp_br1m);
Tbr1delta = ((Qbrf*Sbr1)/Cp_br1m);

%Use built-in MATLAB function 'roots(p)' to find zero(s) of governing function
%and display all the results as well as the final (Mdot) value assumed.

poly_c1   = (Kinlet/(2*rho_br1m*(Acbrf^2)) + (Kbrf/(2*(Acbrf^2)))*(1/rho_br2m ...
    + 1/rho_br3m + 1/rho_br4m + 1/rho_br5m) + (Kbrp/(2*(Acbrp^2)))*(1/rho_br6m ...
    + 1/rho_br7m + 1/rho_br8m) + (Kgsf/(2*(Acgsf^2)))*(1/rho_gs1m + 1/rho_gs2m ...
    + 1/rho_gs3m + 1/rho_gs4m + 1/rho_gs5m) + (Kgsp/(2*(Acgsp^2)))*(1/rho_gs6m ...
    + 1/rho_gs7m));
poly_c2   = ((Slaminlet*Lbr*mu_br1m)/(2*(Dbrf^2)*rho_br1m*Acbrf) ...
```

```
    + ((Slambrf*Lbr)/(2*(Dbrf^2)*Acbrf))*(mu_br2m/rho_br2m ...
    + mu_br3m/rho_br3m + mu_br4m/rho_br4m + mu_br5m/rho_br5m) ...
    + ((Slambrp*Lbr)/(2*(Dbrp^2)*Acbrp))*(mu_br6m/rho_br6m + mu_br7m/rho_br7m ...
    + mu_br8m/rho_br8m) + ((Slamgsf*Lgs)/(2*(Dgsf^2)*Acgsf))*(mu_gs1m/rho_gs1m ...
    + mu_gs2m/rho_gs2m + mu_gs3m/rho_gs3m + mu_gs4m/rho_gs4m + mu_gs5m/rho_gs5m) ...
    + ((Slamgsp*Lgs)/(2*(Dgsp^2)*Acgsp))*(mu_gs6m/rho_gs6m + mu_gs7m/rho_gs7m));
poly_c3   = 0;
poly_c4   = ((-1*g*rho_not)*((Lbr*Tbr1delta)/Tbr1m + (Lgs*Tgs1delta)/Tgs1m ...
    + (Lbr*Tbr2delta)/Tbr2m + (Lgs*Tgs2delta)/Tgs2m + (Lbr*Tbr3delta)/Tbr3m ...
    + (Lgs*Tgs3delta)/Tgs3m + (Lbr*Tbr4delta)/Tbr4m + (Lgs*Tgs4delta)/Tgs4m ...
    + (Lbr*Tbr5delta)/Tbr5m + (Lgs*Tgs5delta)/Tgs5m + (Lbr*Tbr6delta)/Tbr6m ...
    + (Lgs*Tgs6delta)/Tgs6m + (Lbr*Tbr7delta)/Tbr7m + (Lgs*Tgs7delta)/Tgs7m ...
    + (Lbr*Tbr8delta)/Tbr8m));
poly_call = [poly_c1 poly_c2 poly_c3 poly_c4];
root_ans  = (real(roots(poly_call)))';
num_root  = length(root_ans);
I         = find(root_ans > 0);
for K = 1:length(I),
    V(K) = root_ans(I(K));
end;
if (length(V) > 1),
    for K = 1:length(V),
        V_diff(K) = abs(V(K) - Mdot_initial);
    end;
    [Y,I]         = min(V_diff);
    Mdot_root     = V(I);
else,
    Mdot_root     = V(1);
end;
list_root = 1:num_root;
ans_out   = [list_root' root_ans'];
disp(' ');
disp('--------Sub-Calculation Results Using [roots(p)] Function-------');
disp(' ');
disp('   Ans_Num   Root_Result                                          ');
disp(ans_out);
disp(' ');
fprintf ('The (Mdot) result taken from [roots(p)] (kg/s) = %g\n', Mdot_root);
disp(' ');
disp('-------------End Sub-Calculation Results Listing----------------');
disp(' ');
```

< PROPERTYBUOYANCY.M >

```
function [rho_m_ans, Cp_m_ans, mu_m_ans, nu_m_ans] = propertybuoyancy

%M-file function propertybuoyancy.m which determines the mean air density,
%specific heat capacity, dynamic viscosity, and kinematic viscosity based on a
%given mean air temperature input value from the M-file script buoyancy.m.  The
%mean air properties calculated by this function are intended to be used in
%conjunction with buoyancy.m script and massflowbuoyancy.m function M-files.

%Declare static global variables needed within this function or sub-functions
%called by this function.

global Patm;
global Rgas;

%Declare varying global variables needed within this function or sub-functions
%called by this function.

global Tmprop;

%Calculate mean air property values based on mean air temperature input from
%M-file script buoyancy.m.

if (Tmprop >= 250 & Tmprop <= 1300),
    rho_m_ans = Patm/(Rgas*Tmprop);
    %Density of air flow at mean temperature (kg/m^3)
    Cp_m_ans  = 1.057030e+03 - 3.898717e-01*(Tmprop) + 7.746138e-04*(Tmprop^2) ...
        + 4.126734e-08*(Tmprop^3) - 6.788167e-10*(Tmprop^4) + 4.022778e-13*(Tmprop^5) ...
        - 6.529742e-17*(Tmprop^6);
    %Specific heat capacity of air flow at mean temperature (J/kg-K)
    mu_m_ans  = -1.419183e-06 + 9.186130e-08*(Tmprop) - 1.179657e-10*(Tmprop^2) ...
        + 1.349660e-13*(Tmprop^3) - 9.386332e-17*(Tmprop^4) + 3.303345e-20*(Tmprop^5) ...
        - 4.168704e-24*(Tmprop^6);
    %Dynamic viscosity of air flow at mean temperature (kg/m-s)
    nu_m_ans  = (mu_m_ans/rho_m_ans);
    %Kinematic viscosity of air flow at mean temperature (m^2/s)
elseif (Tmprop > 1300 & Tmprop <= 1400),
    rho_m_ans = Patm/(Rgas*Tmprop);
    %Density of air flow at mean temperature (kg/m^3)
    Cp_m_ans  = 1.057030e+03 - 3.898717e-01*(Tmprop) + 7.746138e-04*(Tmprop^2) ...
        + 4.126734e-08*(Tmprop^3) - 6.788167e-10*(Tmprop^4) + 4.022778e-13*(Tmprop^5) ...
        - 6.529742e-17*(Tmprop^6);
    %Specific heat capacity of air flow at mean temperature (J/kg-K)
    mu_m_ans  = -8.361662e-04 + 2.585218e-06*(Tmprop) - 3.169364e-09*(Tmprop^2) ...
        + 2.071844e-12*(Tmprop^3) - 7.533591e-16*(Tmprop^4) + 1.443570e-19*(Tmprop^5) ...
        - 1.138021e-23*(Tmprop^6);
    %Dynamic viscosity of air flow at mean temperature (kg/m-s)
    nu_m_ans  = (mu_m_ans/rho_m_ans);
    %Kinematic viscosity of air flow at mean temperature (m^2/s)
elseif (Tmprop > 1400 & Tmprop <= 3200),
    rho_m_ans = Patm/(Rgas*Tmprop);
    %Density of air flow at mean temperature (kg/m^3)
    Cp_m_ans  = -2.169285e+04 + 6.878214e+01*(Tmprop) - 8.602629e-02*(Tmprop^2) ...
        + 5.716387e-05*(Tmprop^3) - 2.119809e-08*(Tmprop^4) + 4.142479e-12*(Tmprop^5) ...
        - 3.301685e-16*(Tmprop^6);
    %Specific heat capacity of air flow at mean temperature (J/kg-K)
    mu_m_ans  = -8.361662e-04 + 2.585218e-06*(Tmprop) - 3.169364e-09*(Tmprop^2) ...
        + 2.071844e-12*(Tmprop^3) - 7.533591e-16*(Tmprop^4) + 1.443570e-19*(Tmprop^5) ...
        - 1.138021e-23*(Tmprop^6);
    %Dynamic viscosity of air flow at mean temperature (kg/m-s)
    nu_m_ans  = (mu_m_ans/rho_m_ans);
```

```
    %Kinematic viscosity of air flow at mean temperature (m^2/s)
elseif (Tmprop < 250),
    disp(' ');
    disp('------------------------------------------------------------------');
    disp('WARNING: (Tm) is outside applicable range for air property values!');
    disp('...Air properties will be taken at a temperature of 250 Kelvin!...');
    disp('------------------------------------------------------------------');
    disp(' ');
    Tm_warn   = 250;  %Units are (K)
    rho_m_ans = Patm/(Rgas*Tm_warn);
    %Density of air flow at warning temperature (kg/m^3)
    Cp_m_ans  = 1.057030e+03 - 3.898717e-01*(Tm_warn) + 7.746138e-04*(Tm_warn^2) ...
        + 4.126734e-08*(Tm_warn^3) - 6.788167e-10*(Tm_warn^4) ...
        + 4.022778e-13*(Tm_warn^5) - 6.529742e-17*(Tm_warn^6);
    %Specific heat capacity of air flow at warning temperature (J/kg-K)
    mu_m_ans  = -1.419183e-06 + 9.186130e-08*(Tm_warn) - 1.179657e-10*(Tm_warn^2) ...
        + 1.349660e-13*(Tm_warn^3) - 9.386332e-17*(Tm_warn^4) ...
        + 3.303345e-20*(Tm_warn^5) - 4.168704e-24*(Tm_warn^6);
    %Dynamic viscosity of air flow at warning temperature (kg/m-s)
    nu_m_ans  = (mu_m_ans/rho_m_ans);
    %Kinematic viscosity of air flow at warning temperature (m^2/s)
else,
    disp(' ');
    disp('------------------------------------------------------------------');
    disp('WARNING: (Tm) is outside applicable range for air property values!');
    disp('...Air properties will be taken at a temperature of 3200 Kelvin...');
    disp('------------------------------------------------------------------');
    disp(' ');
    Tm_warn   = 3200;  %Units are (K)
    rho_m_ans = Patm/(Rgas*Tm_warn);
    %Density of air flow at warning temperature (kg/m^3)
    Cp_m_ans  = -2.169285e+04 + 6.878214e+01*(Tm_warn) - 8.602629e-02*(Tm_warn^2) ...
        + 5.716387e-05*(Tm_warn^3) - 2.119809e-08*(Tm_warn^4) ...
        + 4.142479e-12*(Tm_warn^5) - 3.301685e-16*(Tm_warn^6);
    %Specific heat capacity of air flow at warning temperature (J/kg-K)
    mu_m_ans  = -8.361662e-04 + 2.585218e-06*(Tm_warn) - 3.169364e-09*(Tm_warn^2) ...
        + 2.071844e-12*(Tm_warn^3) - 7.533591e-16*(Tm_warn^4) ...
        + 1.443570e-19*(Tm_warn^5) - 1.138021e-23*(Tm_warn^6);
    %Dynamic viscosity of air flow at warning temperature (kg/m-s)
    nu_m_ans  = (mu_m_ans/rho_m_ans);
    %Kinematic viscosity of air flow at warning temperature (m^2/s)
end;
```

A.2 EXAMPLE MODEL RUN

The following appendix segment provides an example model run including the complete program input and output as displayed from within the MATLAB command window interface. Furthermore, the corresponding setup for this example run is identical to the MATLAB program base trial from Section 6.4.1 above with a net heat input rate of 600 watts and accepted defaults for remaining input options. Program execution is initiated by simply typing the script M-file name (without extension) 'buoyancy' at the command window prompt, which is not shown in the example run initial input/output below due to the cleanup commands issued at program startup for clearing the command window, workspace, and stored functions from memory.

< EXAMPLE RUN COMMAND WINDOW >

```
-------------------------------------------------------------------------
INTRODUCTION:  The modeled domain in this program is a prototypical
9x9 BWR fuel assembly in a SFP under a severe accident scenario in
which all coolant has been drained from the pool.  It is assumed that
the net decay heat input rate is balanced at steady-state by the heat
removal rate from the system due to a buoyancy-induced air flow rate
through the domain.  This program model uses an energy balance over
the control volume BWR fuel assembly to obtain a buoyancy-induced mass
flow rate solution.  This model references the BWR fuel assembly
components by three region types: fully-populated inlet bundle region
(termed inlet) if necessary, fully-populated bundle and spacer region
(termed full or fully-populated), and partially-populated bundle and
spacer region (termed partial or partially-populated).  Therefore,
enter all parameter values according to this description (if
applicable to the designated parameter name listed in the program
inputs below).

M-FILE DETAILS:  This program is run by the M-file script called
buoyancy.m, which calls M-file function massflowbuoyancy.m for
approximating (Mdot) zero(s) of the governing function.  The outer
loop of this M-File script successively updates the mean air
properties of the flow and the inner loop solves for the convergence
of mean temperatures (from which the air properties are derived by
calling M-file function propertybuoyancy.m).
-------------------------------------------------------------------------


[Enter variable values and criteria required for solution procedure]


-------------------------------------------------------------------------

Prototypical 9x9 BWR fuel assembly geometric values exist as defaults
for this program.  Use these default values?

Options: 1 = Yes (default), 2 = No

 Enter Selection: 1

-------------------------------------------------------------------------

Ambient / inlet environment variable values upon which experimental
and computational experiments were conducted exist as defaults for
this program.  Use these default values?

Options: 1 = Yes (default), 2 = No

 Enter Selection: 1

-------------------------------------------------------------------------
```

```
 Enter net fuel assembly heat input rate (Qt [W]): 600

-------------------------------------------------------------------------

The net heat input rate for each component in the full or partial
region is based on the total length of all fuel rods in a component
compared to the total length of all fuel rods in the associated region
in which the component resides.  However, the fraction of the net fuel
assembly heat input rate designated to the full or partial region is
adjustable if desired.  The fraction value, based on the total length
of all fuel rods in a region compared to the total length of all fuel
rods in the entire fuel assembly, exists as a default for each region
in this program.  Use these default values?

Options: 1 = Yes (default), 2 = No

 Enter Selection: 1

-------------------------------------------------------------------------

The viscous and form loss coefficients for each component as well as
losses due to flow development in the inlet region, which are based on
CFD computational experiments modeling buoyancy-induced flow in a
simplified model domain having similar geometric component length
scales as a prototypical 9x9 BWR fuel assembly, exist as defaults for
this program.  The defaults may not accurately represent these losses
in an actual 9x9 BWR fuel assembly, but do provide reasonable starting
estimates when lacking specific measurement data.  Caution should be
exercised if using these defaults outside the net fuel assembly heat
input range of 300 to 1200 Watts from which these default values are
based.  Use these default values?

Options: 1 = Yes (default), 2 = No

 Enter Selection: 1

-------------------------------------------------------------------------

The stratification term for each component, based on CFD computational
experiments modeling buoyancy-induced flow in a simplified model
domain having similar geometric component length scales as a
prototypical 9x9 BWR fuel assembly, exist as defaults for this
program.  The defaults may not accurately represent these terms in an
actual 9x9 BWR fuel assembly, but do provide reasonable starting
estimates when lacking specific measurement data.  Caution should be
exercised if using these defaults outside the net fuel assembly heat
input range of 300 to 1200 Watts or when using any input values above
other than the default options; from which the stratification default
values here are based.  Use these default values?

Options: 1 = Yes (default), 2 = No
```

```
 Enter Selection: 1

------------------------------------------------------------------------

Initial estimates of the mean air temperature in each assembly
component and the steady-state mass flow rate through the modeled
domain are required inputs, as well as several criterion values for
the solution iteration process.  If default options (or reasonably
similar values) are chosen for the prior inputs above and the net fuel
assembly heat input rate is within (or near) the range of 300 to 1200
Watts, the existing default values stored in this program for these
input parameters are sufficient for obtaining an accurate solution
at any net fuel assembly heat input rate considered.  Use these
default values?

Options: 1 = Yes (default), 2 = No

 Enter Selection: 1

------------------------------------------------------------------------

The following design-specific and critical input values are stored:

Qt (W)              = 600
Slaminlet ()        = 109.175
Slambrf ()          = 98.8083
Slamgsf ()          = 66.2418
Slambrp ()          = 98.0355
Slamgsp ()          = 43.36
f1 ()               = 0.96
Kinlet ()           = -3.30872
Kbrf ()             = 0.300386
Kgsf ()             = 0.128565
Kbrp ()             = 0.93033
Kgsp ()             = 0.318634
Mdot_initial (kg/s) = 0.00129512


------------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00129512

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0011
    3.0000    0.0011


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00112312

-------------End Sub-Calculation Results Listing----------------
```

```
Previous inner loop estimate of (Mdot) value (kg/s) = 0.00129512
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00112312

(Tbr1m) value from current energy balance (K)    = 326.718
(Tbr1m) value from previous iteration update (K) = 330
(Tbr8m) value from current energy balance (K)    = 772.161
(Tbr8m) value from previous iteration update (K) = 870


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00112312

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0011


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00114983

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00112312
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00114983

(Tbr1m) value from current energy balance (K)    = 325.968
(Tbr1m) value from previous iteration update (K) = 328.359
(Tbr8m) value from current energy balance (K)    = 761.063
(Tbr8m) value from previous iteration update (K) = 821.081


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00114983

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00116725

-------------End Sub-Calculation Results Listing----------------
```

```
Previous inner loop estimate of (Mdot) value (kg/s) = 0.00114983
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00116725

(Tbr1m) value from current energy balance (K)    = 325.497
(Tbr1m) value from previous iteration update (K) = 327.163
(Tbr8m) value from current energy balance (K)    = 754.101
(Tbr8m) value from previous iteration update (K) = 791.072


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00116725

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.0011784

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00116725
Current inner loop estimate of (Mdot) value (kg/s)  = 0.0011784

(Tbr1m) value from current energy balance (K)    = 325.203
(Tbr1m) value from previous iteration update (K) = 326.33
(Tbr8m) value from current energy balance (K)    = 749.752
(Tbr8m) value from previous iteration update (K) = 772.587


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.0011784

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00118545

-------------End Sub-Calculation Results Listing----------------
```

```
Previous inner loop estimate of (Mdot) value (kg/s) = 0.0011784
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00118545

(Tbr1m) value from current energy balance (K)    = 325.019
(Tbr1m) value from previous iteration update (K) = 325.766
(Tbr8m) value from current energy balance (K)    = 747.043
(Tbr8m) value from previous iteration update (K) = 761.169


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00118545

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00118988

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00118545
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00118988

(Tbr1m) value from current energy balance (K)    = 324.905
(Tbr1m) value from previous iteration update (K) = 325.393
(Tbr8m) value from current energy balance (K)    = 745.358
(Tbr8m) value from previous iteration update (K) = 754.106


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00118988

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119265

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00118988
```

```
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119265

(Tbr1m) value from current energy balance (K)    = 324.835
(Tbr1m) value from previous iteration update (K) = 325.149
(Tbr8m) value from current energy balance (K)    = 744.311
(Tbr8m) value from previous iteration update (K) = 749.732


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119265

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119437

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119265
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119437

(Tbr1m) value from current energy balance (K)    = 324.791
(Tbr1m) value from previous iteration update (K) = 324.992
(Tbr8m) value from current energy balance (K)    = 743.661
(Tbr8m) value from previous iteration update (K) = 747.021


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119437

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119544

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119437
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119544
```

```
(Tbr1m) value from current energy balance (K)    = 324.763
(Tbr1m) value from previous iteration update (K) = 324.891
(Tbr8m) value from current energy balance (K)    = 743.258
(Tbr8m) value from previous iteration update (K) = 745.341


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119544

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119611

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119544
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119611

(Tbr1m) value from current energy balance (K)    = 324.746
(Tbr1m) value from previous iteration update (K) = 324.827
(Tbr8m) value from current energy balance (K)    = 743.008
(Tbr8m) value from previous iteration update (K) = 744.3


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119611

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119652

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119611
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119652
```

```
(Tbr1m) value from current energy balance (K)    = 324.736
(Tbr1m) value from previous iteration update (K) = 324.787
(Tbr8m) value from current energy balance (K)    = 742.853
(Tbr8m) value from previous iteration update (K) = 743.654


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119652

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119678

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119652
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119678

(Tbr1m) value from current energy balance (K)    = 324.729
(Tbr1m) value from previous iteration update (K) = 324.761
(Tbr8m) value from current energy balance (K)    = 742.757
(Tbr8m) value from previous iteration update (K) = 743.254


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119678

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119694

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119678
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119694

(Tbr1m) value from current energy balance (K)    = 324.725
```

```
(Tbr1m) value from previous iteration update (K) = 324.745
(Tbr8m) value from current energy balance (K)    = 742.698
(Tbr8m) value from previous iteration update (K) = 743.005


-----------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119694

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2581
    2.0000   -0.0012
    3.0000    0.0012


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00119704

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119694
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00119704

(Tbr1m) value from current energy balance (K)    = 324.723
(Tbr1m) value from previous iteration update (K) = 324.735
(Tbr8m) value from current energy balance (K)    = 742.661
(Tbr8m) value from previous iteration update (K) = 742.852


-----------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 324.735
Final inner loop br8 air mean temperature (K) = 742.852
Final inner loop br1 mean air velocity (m/s)  = 0.141117
Final inner loop br8 mean air velocity (m/s)  = 0.344462
Final inner loop air mass flow rate (kg/s)    = 0.00119704


-----------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00129512
Current outer property loop (Mdot) value (kg/s)  = 0.00119704


-----------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00119704

--------Sub-Calculation Results Using [roots(p)] Function-------
```

```
   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0013
    3.0000    0.0013


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00132997

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00119704
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00132997

(Tbr1m) value from current energy balance (K)    = 321.701
(Tbr1m) value from previous iteration update (K) = 324.735
(Tbr8m) value from current energy balance (K)    = 703.173
(Tbr8m) value from previous iteration update (K) = 742.852


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00132997

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0013


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00134496

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00132997
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00134496

(Tbr1m) value from current energy balance (K)    = 321.396
(Tbr1m) value from previous iteration update (K) = 323.218
(Tbr8m) value from current energy balance (K)    = 698.617
(Tbr8m) value from previous iteration update (K) = 723.012


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00134496

--------Sub-Calculation Results Using [roots(p)] Function-------
```

```
   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00135437

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00134496
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00135437

(Tbr1m) value from current energy balance (K)    = 321.209
(Tbr1m) value from previous iteration update (K) = 322.307
(Tbr8m) value from current energy balance (K)    = 695.808
(Tbr8m) value from previous iteration update (K) = 710.815


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00135437

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136023

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00135437
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136023

(Tbr1m) value from current energy balance (K)    = 321.093
(Tbr1m) value from previous iteration update (K) = 321.758
(Tbr8m) value from current energy balance (K)    = 694.08
(Tbr8m) value from previous iteration update (K) = 703.312


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136023

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
```

```
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136385

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136023
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136385

(Tbr1m) value from current energy balance (K)    = 321.022
(Tbr1m) value from previous iteration update (K) = 321.426
(Tbr8m) value from current energy balance (K)    = 693.018
(Tbr8m) value from previous iteration update (K) = 698.696


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136385

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136609

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136385
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136609

(Tbr1m) value from current energy balance (K)    = 320.979
(Tbr1m) value from previous iteration update (K) = 321.224
(Tbr8m) value from current energy balance (K)    = 692.366
(Tbr8m) value from previous iteration update (K) = 695.857


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136609

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
```

```
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136746

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136609
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136746

(Tbr1m) value from current energy balance (K)    = 320.952
(Tbr1m) value from previous iteration update (K) = 321.101
(Tbr8m) value from current energy balance (K)    = 691.966
(Tbr8m) value from previous iteration update (K) = 694.111


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136746

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136831

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136746
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136831

(Tbr1m) value from current energy balance (K)    = 320.936
(Tbr1m) value from previous iteration update (K) = 321.027
(Tbr8m) value from current energy balance (K)    = 691.72
(Tbr8m) value from previous iteration update (K) = 693.039


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136831

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
```

```
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136883

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136831
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136883

(Tbr1m) value from current energy balance (K)    = 320.926
(Tbr1m) value from previous iteration update (K) = 320.981
(Tbr8m) value from current energy balance (K)    = 691.57
(Tbr8m) value from previous iteration update (K) = 692.379


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136883

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136914

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136883
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136914

(Tbr1m) value from current energy balance (K)    = 320.92
(Tbr1m) value from previous iteration update (K) = 320.954
(Tbr8m) value from current energy balance (K)    = 691.477
(Tbr8m) value from previous iteration update (K) = 691.975


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136914

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136934

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136914
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136934

(Tbr1m) value from current energy balance (K)    = 320.916
(Tbr1m) value from previous iteration update (K) = 320.937
(Tbr8m) value from current energy balance (K)    = 691.421
(Tbr8m) value from previous iteration update (K) = 691.726


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136934

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2543
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00136946

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136934
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00136946

(Tbr1m) value from current energy balance (K)    = 320.913
(Tbr1m) value from previous iteration update (K) = 320.926
(Tbr8m) value from current energy balance (K)    = 691.386
(Tbr8m) value from previous iteration update (K) = 691.573


----------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 320.926
Final inner loop br8 air mean temperature (K) = 691.573
Final inner loop br1 mean air velocity (m/s)  = 0.158868
Final inner loop br8 mean air velocity (m/s)  = 0.336484
Final inner loop air mass flow rate (kg/s)    = 0.00136946
```

```
-----------------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00119704
Current outer property loop (Mdot) value (kg/s)  = 0.00136946


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00136946

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0014
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00143637

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00136946
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00143637

(Tbr1m) value from current energy balance (K)    = 319.683
(Tbr1m) value from previous iteration update (K) = 320.926
(Tbr8m) value from current energy balance (K)    = 674.895
(Tbr8m) value from previous iteration update (K) = 691.573


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00143637

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00144352

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00143637
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00144352

(Tbr1m) value from current energy balance (K)    = 319.558
```

```
(Tbr1m) value from previous iteration update (K) = 320.305
(Tbr8m) value from current energy balance (K)    = 673.012
(Tbr8m) value from previous iteration update (K) = 683.234


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00144352

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0014


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00144792

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00144352
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00144792

(Tbr1m) value from current energy balance (K)    = 319.482
(Tbr1m) value from previous iteration update (K) = 319.932
(Tbr8m) value from current energy balance (K)    = 671.86
(Tbr8m) value from previous iteration update (K) = 678.123


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00144792

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145062

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00144792
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145062

(Tbr1m) value from current energy balance (K)    = 319.435
(Tbr1m) value from previous iteration update (K) = 319.707
```

```
(Tbr8m) value from current energy balance (K)    = 671.157
(Tbr8m) value from previous iteration update (K) = 674.992


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145062

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145228

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145062
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145228

(Tbr1m) value from current energy balance (K)    = 319.407
(Tbr1m) value from previous iteration update (K) = 319.571
(Tbr8m) value from current energy balance (K)    = 670.727
(Tbr8m) value from previous iteration update (K) = 673.074


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145228

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145329

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145228
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145329

(Tbr1m) value from current energy balance (K)    = 319.389
(Tbr1m) value from previous iteration update (K) = 319.489
(Tbr8m) value from current energy balance (K)    = 670.465
```

```
(Tbr8m) value from previous iteration update (K) = 671.901


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145329

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145391

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145329
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145391

(Tbr1m) value from current energy balance (K)    = 319.379
(Tbr1m) value from previous iteration update (K) = 319.439
(Tbr8m) value from current energy balance (K)    = 670.305
(Tbr8m) value from previous iteration update (K) = 671.183


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145391

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145429

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145391
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145429

(Tbr1m) value from current energy balance (K)    = 319.372
(Tbr1m) value from previous iteration update (K) = 319.409
(Tbr8m) value from current energy balance (K)    = 670.207
(Tbr8m) value from previous iteration update (K) = 670.744
```

```
-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145429

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145452

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145429
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145452

(Tbr1m) value from current energy balance (K)    = 319.368
(Tbr1m) value from previous iteration update (K) = 319.39
(Tbr8m) value from current energy balance (K)    = 670.148
(Tbr8m) value from previous iteration update (K) = 670.475


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145452

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145466

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145452
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145466

(Tbr1m) value from current energy balance (K)    = 319.366
(Tbr1m) value from previous iteration update (K) = 319.379
(Tbr8m) value from current energy balance (K)    = 670.111
(Tbr8m) value from previous iteration update (K) = 670.311
```

```
-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145466

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2527
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00145475

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145466
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00145475

(Tbr1m) value from current energy balance (K)    = 319.364
(Tbr1m) value from previous iteration update (K) = 319.372
(Tbr8m) value from current energy balance (K)    = 670.089
(Tbr8m) value from previous iteration update (K) = 670.211


-----------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 319.372
Final inner loop br8 air mean temperature (K) = 670.211
Final inner loop br1 mean air velocity (m/s)  = 0.166782
Final inner loop br8 mean air velocity (m/s)  = 0.332766
Final inner loop air mass flow rate (kg/s)    = 0.00145475


-----------------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00136946
Current outer property loop (Mdot) value (kg/s)  = 0.00145475


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00145475

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
```

```
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00148489

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00145475
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00148489

(Tbr1m) value from current energy balance (K)    = 318.859
(Tbr1m) value from previous iteration update (K) = 319.372
(Tbr8m) value from current energy balance (K)    = 663.248
(Tbr8m) value from previous iteration update (K) = 670.211


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00148489

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00148804

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00148489
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00148804

(Tbr1m) value from current energy balance (K)    = 318.808
(Tbr1m) value from previous iteration update (K) = 319.116
(Tbr8m) value from current energy balance (K)    = 662.468
(Tbr8m) value from previous iteration update (K) = 666.73


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00148804

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00148996

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00148804
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00148996

(Tbr1m) value from current energy balance (K)    = 318.776
(Tbr1m) value from previous iteration update (K) = 318.962
(Tbr8m) value from current energy balance (K)    = 661.992
(Tbr8m) value from previous iteration update (K) = 664.599


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00148996

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149114

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00148996
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149114

(Tbr1m) value from current energy balance (K)    = 318.757
(Tbr1m) value from previous iteration update (K) = 318.869
(Tbr8m) value from current energy balance (K)    = 661.702
(Tbr8m) value from previous iteration update (K) = 663.295


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149114

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149185

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149114
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149185

(Tbr1m) value from current energy balance (K)    = 318.745
(Tbr1m) value from previous iteration update (K) = 318.813
(Tbr8m) value from current energy balance (K)    = 661.526
(Tbr8m) value from previous iteration update (K) = 662.499


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149185

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149229

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149185
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149229

(Tbr1m) value from current energy balance (K)    = 318.738
(Tbr1m) value from previous iteration update (K) = 318.779
(Tbr8m) value from current energy balance (K)    = 661.419
(Tbr8m) value from previous iteration update (K) = 662.013


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149229

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149256

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149229
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149256

(Tbr1m) value from current energy balance (K)    = 318.734
(Tbr1m) value from previous iteration update (K) = 318.759
(Tbr8m) value from current energy balance (K)    = 661.353
(Tbr8m) value from previous iteration update (K) = 661.716


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149256

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149272

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149256
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149272

(Tbr1m) value from current energy balance (K)    = 318.731
(Tbr1m) value from previous iteration update (K) = 318.746
(Tbr8m) value from current energy balance (K)    = 661.313
(Tbr8m) value from previous iteration update (K) = 661.534


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149272

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2522
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00149282
```

```
-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149272
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00149282

(Tbr1m) value from current energy balance (K)    = 318.73
(Tbr1m) value from previous iteration update (K) = 318.739
(Tbr8m) value from current energy balance (K)    = 661.289
(Tbr8m) value from previous iteration update (K) = 661.424


---------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 318.739
Final inner loop br8 air mean temperature (K) = 661.424
Final inner loop br1 mean air velocity (m/s)  = 0.170318
Final inner loop br8 mean air velocity (m/s)  = 0.330927
Final inner loop air mass flow rate (kg/s)    = 0.00149282


---------------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00145475
Current outer property loop (Mdot) value (kg/s)  = 0.00149282


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00149282

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150563

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00149282
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150563

(Tbr1m) value from current energy balance (K)    = 318.523
(Tbr1m) value from previous iteration update (K) = 318.739
(Tbr8m) value from current energy balance (K)    = 658.48
```

```
(Tbr8m) value from previous iteration update (K) = 661.424


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150563

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.001507

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150563
Current inner loop estimate of (Mdot) value (kg/s)  = 0.001507

(Tbr1m) value from current energy balance (K)    = 318.501
(Tbr1m) value from previous iteration update (K) = 318.631
(Tbr8m) value from current energy balance (K)    = 658.151
(Tbr8m) value from previous iteration update (K) = 659.952


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.001507

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150783

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.001507
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150783

(Tbr1m) value from current energy balance (K)    = 318.488
(Tbr1m) value from previous iteration update (K) = 318.566
(Tbr8m) value from current energy balance (K)    = 657.951
(Tbr8m) value from previous iteration update (K) = 659.051
```

```
-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150783

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150833

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150783
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150833

(Tbr1m) value from current energy balance (K)    = 318.48
(Tbr1m) value from previous iteration update (K) = 318.527
(Tbr8m) value from current energy balance (K)    = 657.829
(Tbr8m) value from previous iteration update (K) = 658.501


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150833

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150864

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150833
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150864

(Tbr1m) value from current energy balance (K)    = 318.475
(Tbr1m) value from previous iteration update (K) = 318.504
(Tbr8m) value from current energy balance (K)    = 657.755
(Tbr8m) value from previous iteration update (K) = 658.165
```

```
-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150864

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150882

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150864
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150882

(Tbr1m) value from current energy balance (K)    = 318.472
(Tbr1m) value from previous iteration update (K) = 318.489
(Tbr8m) value from current energy balance (K)    = 657.71
(Tbr8m) value from previous iteration update (K) = 657.96


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150882

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2521
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00150894

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150882
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00150894

(Tbr1m) value from current energy balance (K)    = 318.47
(Tbr1m) value from previous iteration update (K) = 318.481
(Tbr8m) value from current energy balance (K)    = 657.683
(Tbr8m) value from previous iteration update (K) = 657.835
```

```
------------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 318.481
Final inner loop br8 air mean temperature (K) = 657.835
Final inner loop br1 mean air velocity (m/s)  = 0.171816
Final inner loop br8 mean air velocity (m/s)  = 0.330115
Final inner loop air mass flow rate (kg/s)    = 0.00150894


------------------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00149282
Current outer property loop (Mdot) value (kg/s)  = 0.00150894


------------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00150894

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151424

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00150894
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151424

(Tbr1m) value from current energy balance (K)    = 318.386
(Tbr1m) value from previous iteration update (K) = 318.481
(Tbr8m) value from current energy balance (K)    = 656.536
(Tbr8m) value from previous iteration update (K) = 657.835


------------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151424

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151485

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151424
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151485

(Tbr1m) value from current energy balance (K)    = 318.377
(Tbr1m) value from previous iteration update (K) = 318.434
(Tbr8m) value from current energy balance (K)    = 656.392
(Tbr8m) value from previous iteration update (K) = 657.186


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151485

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151522

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151485
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151522

(Tbr1m) value from current energy balance (K)    = 318.371
(Tbr1m) value from previous iteration update (K) = 318.405
(Tbr8m) value from current energy balance (K)    = 656.304
(Tbr8m) value from previous iteration update (K) = 656.789


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151522

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015
```

```
The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151544

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151522
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151544

(Tbr1m) value from current energy balance (K)    = 318.367
(Tbr1m) value from previous iteration update (K) = 318.388
(Tbr8m) value from current energy balance (K)    = 656.25
(Tbr8m) value from previous iteration update (K) = 656.546


---------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151544

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151558

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151544
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151558

(Tbr1m) value from current energy balance (K)    = 318.365
(Tbr1m) value from previous iteration update (K) = 318.378
(Tbr8m) value from current energy balance (K)    = 656.218
(Tbr8m) value from previous iteration update (K) = 656.398


---------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 318.378
Final inner loop br8 air mean temperature (K) = 656.398
Final inner loop br1 mean air velocity (m/s)  = 0.172432
Final inner loop br8 mean air velocity (m/s)  = 0.329768
Final inner loop air mass flow rate (kg/s)    = 0.00151558


---------------------------------------------------------------
```

```
Previous outer property loop (Mdot) value (kg/s) = 0.00150894
Current outer property loop (Mdot) value (kg/s)  = 0.00151558


------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151558

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151771

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151558
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151771

(Tbr1m) value from current energy balance (K)    = 318.332
(Tbr1m) value from previous iteration update (K) = 318.378
(Tbr8m) value from current energy balance (K)    = 655.759
(Tbr8m) value from previous iteration update (K) = 656.398


------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151771

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151801

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151771
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151801

(Tbr1m) value from current energy balance (K)    = 318.327
(Tbr1m) value from previous iteration update (K) = 318.355
```

```
(Tbr8m) value from current energy balance (K)    = 655.688
(Tbr8m) value from previous iteration update (K) = 656.079


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151801

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151819

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151801
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151819

(Tbr1m) value from current energy balance (K)    = 318.324
(Tbr1m) value from previous iteration update (K) = 318.341
(Tbr8m) value from current energy balance (K)    = 655.645
(Tbr8m) value from previous iteration update (K) = 655.883


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151819

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.0015183

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151819
Current inner loop estimate of (Mdot) value (kg/s)  = 0.0015183

(Tbr1m) value from current energy balance (K)    = 318.322
(Tbr1m) value from previous iteration update (K) = 318.332
(Tbr8m) value from current energy balance (K)    = 655.619
```

```
(Tbr8m) value from previous iteration update (K) = 655.764


-----------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 318.332
Final inner loop br8 air mean temperature (K) = 655.764
Final inner loop br1 mean air velocity (m/s)  = 0.172687
Final inner loop br8 mean air velocity (m/s)  = 0.32964
Final inner loop air mass flow rate (kg/s)    = 0.0015183


-----------------------------------------------------------------

Previous outer property loop (Mdot) value (kg/s) = 0.00151558
Current outer property loop (Mdot) value (kg/s)  = 0.0015183


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.0015183

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151925

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.0015183
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151925

(Tbr1m) value from current energy balance (K)    = 318.307
(Tbr1m) value from previous iteration update (K) = 318.332
(Tbr8m) value from current energy balance (K)    = 655.417
(Tbr8m) value from previous iteration update (K) = 655.764


-----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151925

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
```

```
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151941

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151925
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151941

(Tbr1m) value from current energy balance (K)    = 318.305
(Tbr1m) value from previous iteration update (K) = 318.32
(Tbr8m) value from current energy balance (K)    = 655.378
(Tbr8m) value from previous iteration update (K) = 655.59


----------------------------------------------------------------

Starting value of (Mdot) for inner loop (kg/s) = 0.00151941

--------Sub-Calculation Results Using [roots(p)] Function-------

   Ans_Num   Root_Result
    1.0000   -0.2520
    2.0000   -0.0015
    3.0000    0.0015


The (Mdot) result taken from [roots(p)] (kg/s) = 0.00151951

-------------End Sub-Calculation Results Listing----------------


Previous inner loop estimate of (Mdot) value (kg/s) = 0.00151941
Current inner loop estimate of (Mdot) value (kg/s)  = 0.00151951

(Tbr1m) value from current energy balance (K)    = 318.303
(Tbr1m) value from previous iteration update (K) = 318.312
(Tbr8m) value from current energy balance (K)    = 655.355
(Tbr8m) value from previous iteration update (K) = 655.484


----------------------------------------------------------------

Inner loop (Tm) iterations have successfully converged!

Final inner loop br1 air mean temperature (K) = 318.312
Final inner loop br8 air mean temperature (K) = 655.484
Final inner loop br1 mean air velocity (m/s)  = 0.172799
Final inner loop br8 mean air velocity (m/s)  = 0.329582
```

```
Final inner loop air mass flow rate (kg/s)    = 0.00151951

-----------------------------------------------------------------

Outer air properties update loop has converged at (Tm) values!

-----------------------------------------------------------------

Final Model Results:

Final assembly air mass flow rate (kg/s) = 0.00151951
Final assembly exit air temperature (K)  = 677.628
Final gs7 exit air temperature (K)       = 636.821
Final br7 exit air temperature (K)       = 634.181
Final gs6 exit air temperature (K)       = 592.975
Final br6 exit air temperature (K)       = 590.31
Final gs5 exit air temperature (K)       = 548.727
Final br5 exit air temperature (K)       = 545.711
Final gs4 exit air temperature (K)       = 498.638
Final br4 exit air temperature (K)       = 495.593
Final gs3 exit air temperature (K)       = 448.085
Final br3 exit air temperature (K)       = 445.013
Final gs2 exit air temperature (K)       = 397.13
Final br2 exit air temperature (K)       = 394.037
Final gs1 exit air temperature (K)       = 345.864
Final br1 exit air temperature (K)       = 342.756
Final br8 mean air temperature (K)       = 655.484
Final gs7 mean air temperature (K)       = 635.65
Final br7 mean air temperature (K)       = 611.724
Final gs6 mean air temperature (K)       = 591.78
Final br6 mean air temperature (K)       = 569.354
Final gs5 mean air temperature (K)       = 547.212
Final br5 mean air temperature (K)       = 522.023
Final gs4 mean air temperature (K)       = 497.099
Final br4 mean air temperature (K)       = 471.651
Final gs3 mean air temperature (K)       = 446.532
Final br3 mean air temperature (K)       = 420.84
Final gs2 mean air temperature (K)       = 395.566
Final br2 mean air temperature (K)       = 369.665
Final gs1 mean air temperature (K)       = 344.293
Final br1 mean air temperature (K)       = 318.312
Final br8 mean air velocity (m/s)        = 0.329582
Final gs7 mean air velocity (m/s)        = 0.405304
Final br7 mean air velocity (m/s)        = 0.307573
Final gs6 mean air velocity (m/s)        = 0.377323
Final br6 mean air velocity (m/s)        = 0.286263
Final gs5 mean air velocity (m/s)        = 0.396131
Final br5 mean air velocity (m/s)        = 0.283467
Final gs4 mean air velocity (m/s)        = 0.35984
Final br4 mean air velocity (m/s)        = 0.256103
Final gs3 mean air velocity (m/s)        = 0.323219
Final br3 mean air velocity (m/s)        = 0.228499
```

```
Final gs2 mean air velocity (m/s)        = 0.286309
Final br2 mean air velocity (m/s)        = 0.200697
Final gs1 mean air velocity (m/s)        = 0.249174
Final br1 mean air velocity (m/s)        = 0.172799

Final total pressure loss across assembly (Pa) = 14.1979


---------------------------------------------------------------------
End Model Run
---------------------------------------------------------------------
```

References

Aglar, F., Gelisli, M. O., and Ozturk, E. (2007). Keeping It Cool. *ANSYS Advantage , I* (2), 24-25.

Alexandrou, A. (2001). *Principles of Fluid Mechanics.* Upper Saddle River: Prentice-Hall.

Alvarez, R., Beyea, J., Janberg, K., Kang, J., Lyman, E., MacFarlane, A., *et al.* (2003). Reducing the Hazards from Stored Spent Power-Reactor Fuel in the United States. *Science and Global Security , 11* (1), 1-51.

Anderson, J. D. (1995). *Computational Fluid Dynamics.* New York: McGraw-Hill, Inc.

Armaly, B. F., Durst, F., Pereira, J. C., and Schonung, B. (1983). Experimental and Theoretical Investigation of Backward-Facing Step Flow. *Journal of Fluid Mechanics* , 473-496.

Artnak, E. J. (2008). *Assessment of Radiation Heat Transfer Models Within FLUENT Flow Modeling Software in Corroboration of MELCOR COR Package Models and Calibration Experiments: A Qualitative Approach.* Thesis, The University of Texas at Austin, Department of Mechanical Engineering.

Artnak, E. J., Biegalski, S. R., Howell, J. R., Gauntt, R., and Wagner, K. C. (2005). Benchmarking the MELCOR Radiative Heat Transfer Model with FLUENT. *Transactions of the American Nuclear Society , 93*, 681-682.

Ball, K. (2004). Grid Generation in FLUENT. *ME 369L Course Notes* . The University of Texas at Austin.

Ball, K. (2004). Solution of Navier-Stokes Equations. *ME 369L Course Notes* . The University of Texas at Austin.

Ball, K. (2004). Turbulence Models in FLUENT. *ME 369L Course Notes* . The University of Texas at Austin.

Bell, B. (2004). Case Studies on Using the Turbulence Models in FLUENT 6. *2004 CFD Summit*, (pp. 1-76).

Benjamin, A. S., McCloskey, D. S., Powers, D. A., and Dupree, S. A. (1979). *Spent Fuel Heatup Following Loss of Water During Storage.* Sandia National Laboratories. NUREG/CR-0649.

Best, F., and Pisano, N. (1983). The Propagation of Self-Sustaining Zirconium Oxidation in a Drained Spent Fuel Storage Pool. *Transactions of the American Nuclear Society , 45*, 133-134.

Bhaskaran, R., and Collins, L. (2002). Introduction to CFD Basics. *FLUENT Tutorials Short Course Notes* . Ithaca, New York: Unpublished manuscript, Cornell University.

Bird, R. B., Stewart, W. E., and Lightfoot, E. N. (1960). *Transport Phenomena.* New York: John Wiley & Sons, Inc.

Brinkworth, B. J. (2000). Estimation of Flow and Heat Transfer for the Design of PV Cooling Ducts. *Solar Energy , 69* (5), 413-420.

Brinkworth, B. J., and Sandberg, M. (2005). A Validated Procedure for Determining the Buoyancy-Induced Flow in Ducts. *Building Services Engineering Research and Technology , 26* (1), 35-48.

Brinkworth, B. J., Cross, B. M., Marshall, R. H., and Yang, H. (1997). Thermal Regulation of Photovoltaic Cladding. *Solar Energy , 61* (3), 169-178.

Brinkworth, B. J., Marshall, R. H., and Ibarahim, Z. (2000). A Validated Model of Naturally Ventilated PV Cladding. *Solar Energy , 69* (1), 67-81.

CD-adapco. (2000, May 17). Nuclear Reactor Core Cooling Simulations Using STAR-CD. *SP World: Nuclear Applications Presentation* . Argonne National Laboratory collaborative research.

Chilka, A., and Kulkarni, A. (n.d.). *Modeling Turbulent Flows in FLUENT.* Unpublished manuscript.

Collins, T. E., and Hubbard, G. (2001). *Technical Study of Spent Fuel Pool Accident Risk at Decommissioning Nuclear Power Plants.* NUREG-1738.

Conner, M. E., Baglietto, E., and Elmahdi, A. M. (2010). CFD Methodology and Validation for Single-Phase Flow in PWR Fuel Assemblies. *Nuclear Engineering and Design, In press.*

Cruchaga, M., and Celentano, D. (2003). Modelling Natural and Mixed Convection in Obstructed Channels. *International Journal of Numerical Methods for Heat and Fluid Flow , 13* (1), 57-85.

Durbin, S., and Lindgren, E. (2005). *Hydraulic Analysis of the Spent Fuel Pool Experiment.* Sandia National Laboratories, Experiments and New Programs, Albuquerque.

FLUENT Inc. (n.d.). *A Brief History of CFD.* Retrieved 2010, from http://www.fluent.com/about/cfdhistory.htm

FLUENT Inc. (2006, September). FLUENT 6.3 Tutorial Guide.

FLUENT Inc. (2006, September). FLUENT 6.3 User's Guide.

FLUENT Inc. (2006, September). FLUENT 6.3 Validation Guide.

FLUENT Inc. (2005). *FLUENT Incorporated*. Retrieved 2005, from FLUENT Incorporated Web site: http://www.fluent.com

FLUENT Inc. (2006, March). GAMBIT 2.3 Modeling Guide.

FLUENT Inc. (2006, March). GAMBIT 2.3 User Defined Function Guide.

FLUENT Inc. (2006, March). GAMBIT 2.3 User's Guide.

FLUENT Inc. (2005, April). Modeling Turbulent Flows with FLUENT. *Advanced FLUENT Training: Turbulence* , 1-11.

FLUENT Inc. (2005, April). Overview of Turbulence Modeling. *Advanced FLUENT Training: Turbulence* , 1-65.

FLUENT Inc. (2006, May). TGrid 4.0 Tutorial Guide.

FLUENT Inc. (2006, May). TGrid 4.0 User's Guide.

FLUENT Inc. (2005, April). Turbulence Modeling Options in FLUENT. *Advanced FLUENT Training: Turbulence* , 1-53.

Fox, R. W., and McDonald, A. T. (1985). *Introduction to Fluid Mechanics* (3rd ed.). New York: John Wiley & Sons.

Fox, R. W., and McDonald, A. T. (1992). *Introduction to Fluid Mechanics* (4th ed.). New York: John Wiley & Sons.

Freitas, C. J. (1993). Journal of Fluids Engineering Editorial Policy Statement on the Control of Numerical Accuracy. *ASME Journal of Fluids Engineering , 115*, 339-340.

Gajapathy, R., Velusamy, K., Selvaraj, P., Chellapandi, P., and Chetal, S. C. (2007). CFD Investigation of Helical Wire-Wrapped 7-Pin Fuel Bundle and the Challenges in Modeling Full Scale 217-Pin Bundle. *Nuclear Engineering and Design , 237* (24), 2332-2342.

Gauntt, R. O., and Wagner, K. C. (2004). *Evaluation of a BWR Spent Fuel Pool Accident Response to Loss-of-Coolant Inventory Scenarios Using MELCOR 1.8.5.* Rev. 1, Draft, Sandia National Laboratories.

Gauntt, R. O., Cole, R. K., Erickson, C. M., Gido, R. G., Gasser, R. D., Rodriquez, S. B., *et al.* (2000, May). MELCOR Computer Code Manuals Version 1.8.5. *NUREG/CR-6119* . Albuquerque, NM: Sandia National Laboratories.

Gay, R. R. (1984). Spent Nuclear Fuel Storage Pool Thermal Hydraulic Analysis. *Progress in Nuclear Energy , 14* (3), 199-225.

Gotovskii, M. A., Fedorovich, E. D., Fromzel, V. N., and Shleifer, V. A. (1984). Heat Transfer of a Vertical Bundle of Heat-Releasing Rods in the Absence of Circulation of the Heat Carrier. *Inzhenerno-Fizicheskii Zhurnal , 46* (4), 549-554.

Hamman, K. D., and Berry, R. A. (2009). A CFD Simulation Process for Fast Reactor Fuel Assemblies. *Nuclear Engineering and Design, In press.*

Haslinger, K. H., Joffre, P. F., Rossano, F. V., Nilsson, S., and Norstrom, L. (1997, August 17-22). Qualification Testing of PWR Fuel Assemblies at ABB. *Transactions of the 14th International Conference on Structural Mechanics in Reactor Technology* , 157-164. Lyon, France.

Hohne, T., Kliem, S., and Bieder, U. (2006). Modeling of a Buoyancy-Driven Flow Experiment at the ROCOM Test Facility Using the CFD Codes CFX-5 and Trio_U. *Nuclear Engineering and Design , 236* (12), 1309-1325.

Holman, J. P. (1990). *Heat Transfer* (7th ed.). New York: McGraw-Hill.

Iaccarino, G. (2004). Solution Methods for the Incompressible Navier-Stokes Equations. *ME469B/3/GI Course Notes* . Unpublished manuscript, Stanford University.

Ikeda, K., Makino, Y., and Hoshi, M. (2006). Single-Phase CFD Applicability for Estimating Fluid Hot-Spot Locations in a 5x5 Fuel Rod Bundle. *Nuclear Engineering and Design , 236* (11), 1149-1154.

Incropera, F. P., and DeWitt, D. P. (2002). *Fundamentals of Heat and Mass Transfer* (5th ed.). New York: John Wiley & Sons.

Jeong, J. H., and Han, B.-S. (2008). Coolant Flow Field in a Real Geometry of PWR Downcomer and Lower Plenum. *Annals of Nuclear Energy , 35* (4), 610-619.

Lee, J. C., Choi, W. S., Bang, K. S., Seo, K. S., and Yoo, S. Y. (2009). Thermal-Fluid Flow Analysis and Demonstration Test of a Spent Fuel Storage System. *Nuclear Engineering and Design , 239* (3), 551-558.

Lindgren, E. (2004). *Spent Fuel Pool Heatup and Propagation Phenomena Experiments.* Draft Preliminary Test Plan, Sandia National Laboratories.

Liu, C. C., and Ferng, Y. M. (2010). Numerically Simulating the Thermal-Hydraulic Characteristics Within the Fuel Rod Bundle Using CFD Methodology. *Nuclear Engineering and Design, In press.*

Lombardo, N. J., Cuta, J. M., Michener, T. E., Rector, D. R., and Wheeler, C. L. (1986). *COBRA-SFS: A Thermal-Hydraulic Analysis Computer Code Volume III: Validation Assessments.* Pacific Northwest National Laboratory. Richland: PNL-6049.

Lorenzini, E., Sola, P. G., and Spiga, M. (1983). Radiative Heat Transfer in a Pressurised Water Reactor Fuel Bundle Under Accident Conditions. *Nuclear Technology , 63*, 180-184.

Mandell, D. A. (1979). *A Radiative Heat Transfer Model for the TRAC Code.* Los Alamos Scientific Laboratory. LA-7965-MS.

Mandell, D. A. (1981). Geometrical View Factors For Radiative Heat Transfer Within Boiling Water Reactor Fuel Bundles. *Nuclear Technology , 52*, 383-392.

Mbaye, M., and Bilgen, E. (1992). Natural Convection and Conduction in Porous Wall, Solar Collector Systems Without Vents. *Transactions of the ASME , 114*, 40-46.

Muneer, T., and Han, B. (1996). Simplified Analysis for Free Convection in Enclosures - Application to an Industrial Problem. *Energy Conversion and Management , 37* (9), 1463-1467.

Naitoh, M., Kawabe, R., and Chino, K. (1977). Analysis of Radiant Heat Transfer in a BWR Fuel Assembly. *Nuclear Engineering and Design , 44* (3), 315-321.

Natesan, K., Sundararajan, T., Narasimhan, A., and Velusamy, K. (2010). Turbulent Flow Simulation in a Wire-Wrap Rod Bundle of an LMFBR. *Nuclear Engineering and Design , 240* (5), 1063-1072.

Niyogi, K. K., and Tseng, C. W. (1983). Thermal Hydraulics of a Spent Fuel Pool. *ASME Paper 83-NE-9* .

Patankar, S. V. (1981). A Calculation Procedure for Two-Dimensional Elliptic Situations. *Numerical Heat Transfer, Part A: Applications , 4*, 409-425.

Patankar, S. V., and Spalding, D. B. (1972). A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows. *International Journal of Heat and Mass Transfer , 15*, 1787-1806.

Rector, D. R., Wheeler, C. L., and Lombardo, N. J. (1986). *COBRA-SFS: A Thermal-Hydraulic Analysis Computer Code Volume I: Mathematical Models and Solution Method.* Pacific Northwest National Laboratory. Richland: PNL-6049.

Reinert, D. (2004). *Investigation of Thermal Radiation in Spent Fuel Pools.* The University of Texas at Austin, Department of Mechanical Engineering. Unpublished manuscript.

Renner, H. (1976). SFPT-4 Computer Program for Multi-Channel Analysis of Spent Fuel Pool Natural Circulation. *NUS-TM-285* .

Rhie, C. M., and Chow, W. L. (1983). Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation. *AIAA Journal , 21* (11), 1525-1532.

Roache, P. J., Ghia, K. N., and White, F. M. (1986). Editorial Policy Statement on the Control of Numerical Accuracy. *ASME Journal of Fluids Engineering , 108*, 2.

Rohde, U., Hohne, T., and Kliem, S. (2006). Simulating Turbulent Mixing in Nuclear Reactor Pressure Vessels. *ANSYS Solutions , 7* (2), 27-28.

Singh, K. P., and Soler, A. I. (1983). A Method for Computing the Maximum Water Temperature in a Fuel Pool Containing Spent Nuclear Fuel. *ASME Paper 83-NE-7* .

Smolec, W., and Thomas, A. (1993). Theoretical and Experimental Investigations of Heat Transfer in a Trombe Wall. *Energy Conversion and Management , 34* (5), 385-400.

SolidWorks Corporation. (1995-2002). *Introducing SolidWorks.* Concord, Massachusetts.

SolidWorks Corporation. (2005). *SolidWorks Corporation*. Retrieved 2005, from SolidWorks Corporation Web site: http://www.solidworks.com

Sparrow, E. M., and Loeffler, A. L. (1959). Longitudinal Laminar Flow Between Cylinders Arranged in Regular Array. *AIChE Journal , 5* (3), 325-330.

Special Metals Corporation. (2006, January). *INCONEL Alloy 625: Technical Bulletin.* Retrieved October 2009, from Special Metals Corporation Web site: http://www.specialmetals.com/products/inconelalloy625.php

Spiegel, M. R., and Liu, J. (1999). *Mathematical Handbook of Formulas and Tables* (2nd ed.). New York: McGraw-Hill.

Stalker, K. T., and Benjamin, A. S. (1983). Experimental Studies of Rapidly Oxidizing Zircaloy Cladding Bundles in Air. *Transactions of the American Nuclear Society , 45*, 134-135.

Sun, K. H., Gonzalez-Santalo, J. M., and Tien, C. L. (1976, August). Calculations of Combined Radiation and Convection Heat Transfer in Rod Bundles Under Emergency Cooling Conditions. *ASME Journal of Heat Transfer* , 414-420.

Systat Software Inc. (2011). *TableCurve 2D Automated Curve Fitting Analysis*. Retrieved 2011, from Systat Software Inc. Web site: http://www.sigmaplot.com/products/tablecurve2d/tablecurve2d.php

Tautges, T. J. (n.d.). *MOAB-SD: Integrated Structured and Unstructured Mesh Representation.* Sandia National Laboratories, Albuquerque.

Texas Advanced Computing Center. (2010). *HPC Systems*. Retrieved 2011, from TACC Web site: http://www.tacc.utexas.edu/resources/hpc

The Editorial Board. (n.d.). Policy Statement on Numerical Accuracy. *ASME Journal of Heat Transfer* .

The MathWorks, Inc. (1984-2002). *MATLAB 6.5*. Retrieved 2002, from The MathWorks, Incorporated Web site: http://www.mathworks.com

Tokaty, G. A. (1971). *A History and Philosophy of Fluid Mechanics.* Oxfordshire, England: G. T. Foulis and Co., Ltd., Henley-on-Thames.

Toth, S., and Aszodi, A. (2010). CFD Analysis of Flow Field in a Triangular Rod Bundle. *Nuclear Engineering and Design , 240* (2), 352-363.

Travis, R. J., Davis, R. E., Grove, E. J., and Azarm, M. A. (1997). *A Safety and Regulatory Assessment of Generic BWR and PWR Permanently Shutdown Nuclear Power Plants.* Brookhaven National Laboratory. NUREG/CR-6451.

U.S. Nuclear Regulatory Commission. (Fall 2001). Accident Scenario Modeling of Spent Fuel Bundles. *Articles from FLUENT Newsletters (NL297)* , 1-2. FLUENT Inc.

U.S. Nuclear Regulatory Commission. (2001). Spent Fuel Heat-Up. *Application Briefs from FLUENT (EX155)* , 1-2. FLUENT Inc.

Uchida, M., and Nakamura, J. (1981). Matrix Calculation of Radiant Heat Transfer in LWR Fuel Bundles Under Accident Conditions. *Nuclear Engineering and Design , 65*, 63-69.

Uygur, S., and Egrican, N. (1996). Modeling of Natural Turbulent Flow in a Passively Heated Zone. *Energy Conversion and Management , 37* (5), 505-520.

Versteeg, H. K., and Malalasekera, W. (1995). *An Introduction to Computational Fluid Dynamics: The Finite Volume Method.* Harlow, England: Pearson-Prentice Hall.

Walters, K. D., and Leylek, J. H. (2004). A New Model for Boundary Layer Transition Using a Single-Point RANS Approach. *ASME Journal of Turbomachinery , 126*, 193-202.

Weaver, K. D., MacDonald, P., Schultz, R. R., and Moeykens, S. (2003). Gas Cooled Nuclear Reactors. In A. Shenoy (Ed.), *FLUENT Gas-Cooled Reactor Web-Based Seminar.* INEEL, FLUENT, General Atomics.

Webb, S., and Khalil, I. (2005). *Spent Fuel Pool Modeling.* Informal Report, Sandia National Laboratories.

Wilcox, D. C. (1998). *Turbulence Modeling for CFD* (2nd ed.). La Canada, California: DCW Industries, Inc.

Yang, H. X., Marshall, R. H., and Brinkworth, B. J. (1996). Validated Simulation for Thermal Regulation of Photovoltaic Wall Structures. *Proceedings of the 25th IEEE PV Specialists Conference*, (pp. 1453-1456). Washington D.C.

Young, D. F., Munson, B. R., and Okiishi, T. H. (2004). *A Brief Introduction to Fluid Mechanics* (3rd ed.). Hoboken, NJ: John Wiley & Sons, Inc.

Vita

Edward Joseph Artnak III was born in Bethesda, Maryland on January 20, 1980, the son of Dr. Edward Joseph Artnak Jr. and Dr. Kathryn Elizabeth Artnak. He attended Central High School in San Angelo, Texas, graduating in May of 1999. The following August, he entered The University of Texas at Austin as a Texas Exes Scholarship and Randall & Dewey, Inc. Scholarship recipient.

During his undergraduate career, he gained valuable work experience in the field of mechanical engineering as a product development intern at Ford Motor Company in the summer of 2002 and as a project engineering facilities intern at Occidental Oil & Gas Corporation in the summer of 2003. He was also employed as an undergraduate assistant in the Department of Mechanical Engineering at The University of Texas at Austin during the spring semester of 2004 before receiving his Bachelor of Science degree in mechanical engineering from The University of Texas at Austin in May 2004.

In the summer of 2004, he furthered his work experience in the field of engineering as a product development intern for Fallbrook Technologies and entered the Department of Mechanical Engineering graduate program at The University of Texas at Austin in August 2004. Upon entering the graduate program, he began work as a graduate research assistant within the Nuclear and Radiation Engineering Program conducting research in CFD modeling and analysis of spent nuclear fuel pools as part of a conjoint experimental research effort with Sandia National Laboratories. This includes a period in which he performed on-site research at Sandia National Laboratories as a contractor during the summer of 2005.

He received his Master of Science in Engineering from The University of Texas at Austin in May 2008 and continued work as a doctoral student within the Nuclear and Radiation Engineering program at The University of Texas at Austin in the summer of 2008. The research undertaken in his pursuit of a Doctor of Philosophy in Mechanical Engineering is largely a continuance of initial collaborative research efforts with Sandia National Laboratories to employ computational models in the analysis of spent nuclear fuel pools.

Email address: UT@att.net

This dissertation was typed by the author.